

Contents

EXECUTIVE SUMMARY	1
USECASE BACKGROUND	1
OBJECTIVE	2
BUILDING THE ERP CLOUD PO EVENT USE CASE	2
Prerequisites.....	2
Import VBCS Web Application.....	3
Creating Connections (Flow1)	4
Creating REST Connection	5
Creating the PO Event Integration (Flow1)	7
Activate Integration (Flow1).....	32
Testing the ERP Event Flow Integration (Flow1)	33
Creating Connections (Flow2)	39
Creating the PO Proxy Integration (Flow2)	39
Activate Integration (Flow2).....	50
Invoke ERP Cloud Service from VBCS Web App	51
Running the VBCS Web App	58

EXECUTIVE SUMMARY

Oracle Integration provides native connectivity to Oracle and non-Oracle Software as a Service (SaaS) and On-premises applications, such as Oracle ERP Cloud, Oracle Service Cloud, HCM Cloud, Salesforce.com, Workday, EBS, SAP, NetSuite and so on. Oracle Integration adapters simplify connectivity by handling the underlying complexities of connecting to applications using industry-wide best practices. You only need to create a connection that provides minimal connectivity information for each system.

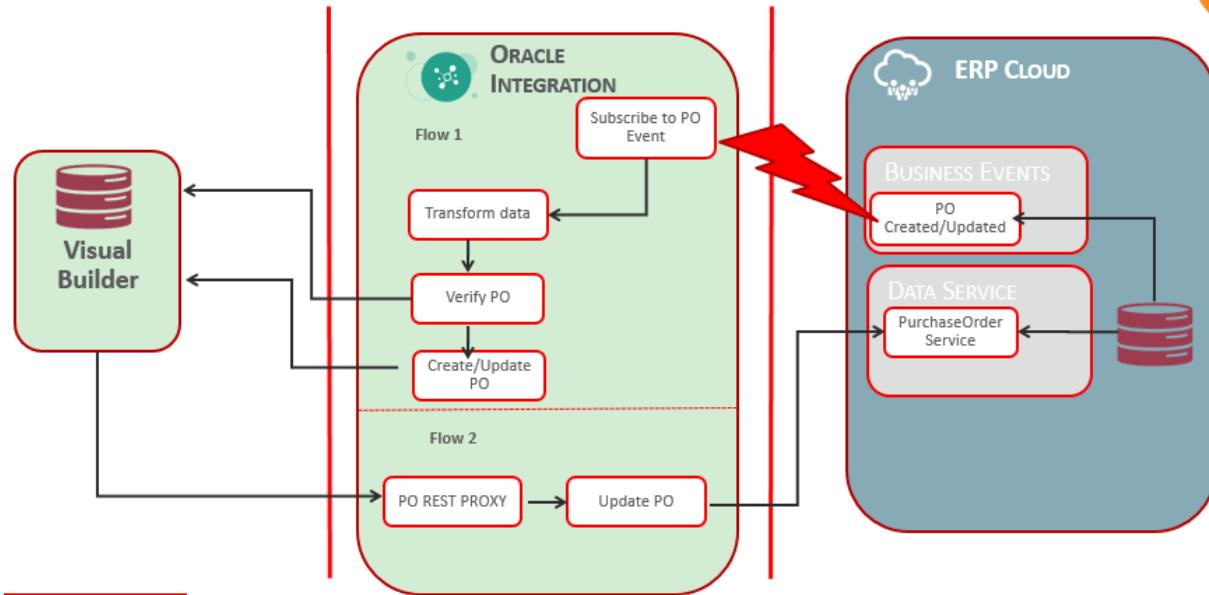
USECASE BACKGROUND

This use case explores the use of Oracle Integration to subscribe to Oracle ERP Cloud Events and push the relevant event information to downstream systems. As part of the lab you would be building the below use case scenario.

- a. User creates a Purchase Order(PO) in ERP Cloud and a PO event is raised (Flow1)
- b. Oracle Integration subscribes to the PO event, transforms and pushes the relevant information to a custom table (VBCS) (Flow1)
- c. User attaches Letter of Credit (LOC) information to the PO record from the VBCS Web App. (Flow2)
- d. PO Record in ERP Cloud is updated with the LOC information in ERP cloud (Flow2)

The following diagram illustrates the proposed interaction between the systems involved in this use case. There 2 flows to complete this use case.

Near Real-Time Synchronization of Purchase Order



OBJECTIVE

This document walks you through the steps needed to replicate this use case in your environment

BUILDING THE ERP CLOUD PO EVENT USE CASE

This section works through the steps that are required to build the integration from scratch.

Prerequisites

You will need access to the following applications and artifacts:

- Oracle Integration (OIC)
- ERP Cloud R12+
- VBCS App to import (LetterOfCreditPortal.zip)

Import VBCS Web Application

1. Login to Oracle Integration homepage and Click on Visual Builder
2. Select Import -> Application from file and Upload the "LetterOfCreditPortal.zip" from the Lab artifacts
Note: Since there are multiple attendees sharing a single environment, please use the `<ClassID> <StudentID>` information (will be provided to you by the instructor) as a suffix to your web application
E.g. If your `<ClassID>` is 96 and your `<StudentID>` is 06, then you will name your web application as **LetterOfCreditPortal9606**
3. Change the ApplicationId also to the same name (**LetterOfCreditPortal9606**) to avoid conflicts and Select **Import**

Import Application from File X

 LetterOfCreditPortal.zip

* Application Name
Best to keep the name short so it looks nice

* Application ID
This ID defines the context path (browser's URI) used for the application

Description

Cancel Import

4. Confirm that your application is imported into VBCS. Open the Imported application and you would notice that a few pages and business objects relevant to the use case is pre created. We will use this web application in the later part of the lab.

Select the imported Web Application and you will observe a few pages are pre created

5. Couple of Business Objects (PO and LOC) pre created

Click the PO Business Object, copy the Data endpoint as highlighted above, and save it for later use.

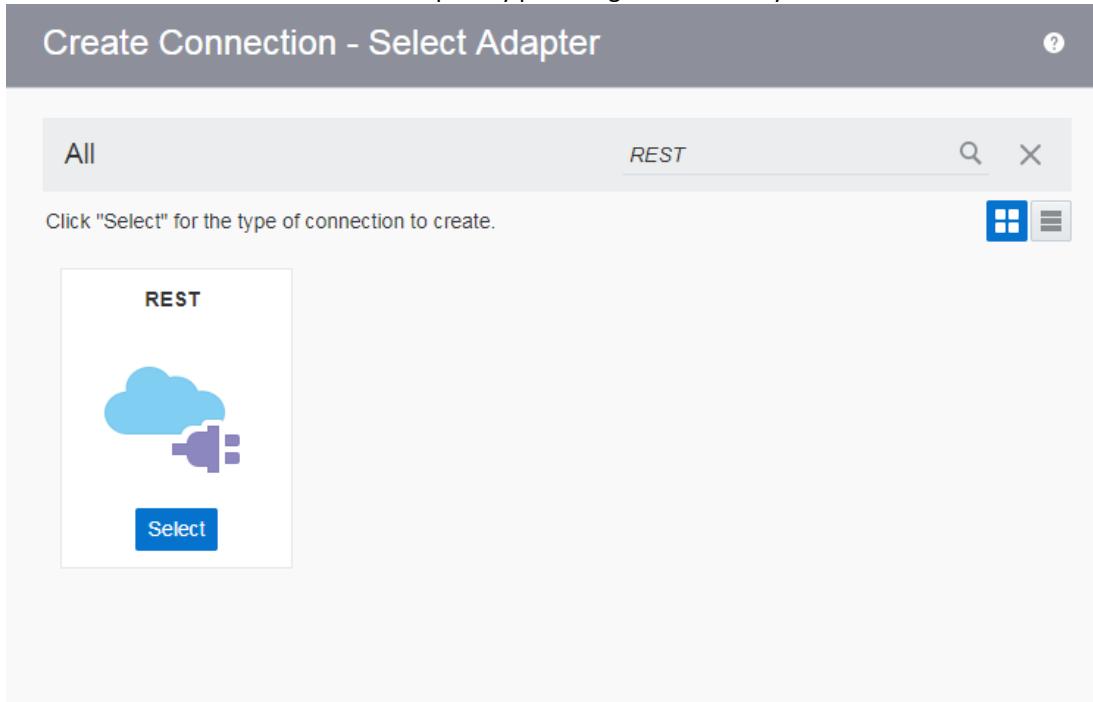
Creating Connections (Flow1)

The following Connections have been created and configured. You will be using these connections for creating Integration flows

Connection Name	Connection Type
ERP Conn 96 06	ERP Adapter

Creating REST Connection

1. On the Oracle Integration home page, click Integrations
2. Select the Connections tab
3. Click on Create and search for REST Adapter by providing the search keyword “REST” and Select Create



4. Provide following details and click on Create

Field	Enter
Name	VBCS REST Con <ClassId><StudentId> Eg: VBCS REST Con 96 06
Identifier	Nothing to enter. The value is automatically generated from the integration name: VBCS_REST_CON_<ClassId>_<StudentId> Eg: VBCS_REST_CON_96_06
Role	Invoke

Create New Connection

Enter information that describes the connection. Use a meaningful name and description to help others find your connection when they create their own integrations. The Identifier must be unique and can be set only when the connection is created.

* Name	VBCS REST Con 96 06
* Identifier	VBCS_REST_CON_96_06
Role	Invoke
Description	Enter a brief description...

Create **Cancel**

5. Configure Connectivity and Security Property Values as per below table and Click "OK"

Field	Enter
<u>Configure Connectivity</u>	
Connection type	REST API Base URL
Connection URL	Use the Data endpoint url copied from "Import VBCS App" Section" -> Step 5 Provide the endpoint till "/data " Ex: <a href="https://<oichost>/ic/builder/design/LetterOfCreditPortal9606/1.0/resources/data">https://<oichost>/ic/builder/design/LetterOfCreditPortal9606/1.0/resources/data
<u>Configure Security</u>	
Security Policy	Basic Authentication
Username	Use the same credentials as used for login into Oracle Integration console
Password	Use the same credentials as used for login into Oracle Integration console

Connection Properties

Enter information so we can connect to your application/endpoint and process requests.

Property Name	Property Value
* Connection Type	REST API Base URL
TLS Version	< Please select an item from the list >
* Connection URL	cpm.uscom-central-1.oraclecloud.com/ic/builder/design/LetterOfCreditPortal9606/1.0/resources/data
Enable two way SSL for outbound connect	< Please select an item from the list >
Identity keystore alias name (Optional)	Name of alias to use for establishing identity during two way SSL communication

OK **Cancel**

Credentials

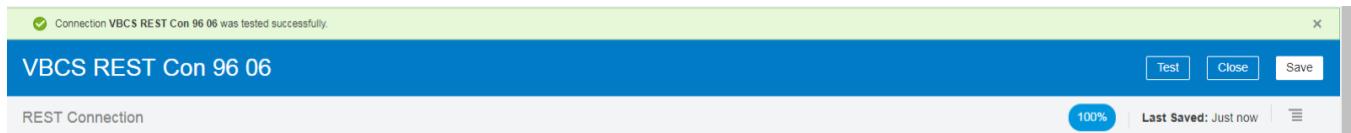
You can configure the Security Policy for this connection. Please select the Security Policy.

Security Policy Basic Authentication ▾

Your application/endpoint requires that users and services provide security credentials for access. Specify the login credentials below.

Property Name	Property Value
* Username	kishore.x.katta@oracle.com
* Password	*****
* Confirm Password	*****

OK **Cancel**

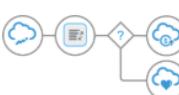
Save and Test the connection**Creating the PO Event Integration (Flow1)**

1. On the Oracle Integration home page, click **Integrations**.
2. On the Integrations page, click **Create**. The Create Integration - Select a Style/Pattern dialog is displayed.
3. Select **App Driven Orchestration** type of integration. The **Create New Integration** dialog is displayed.
4. Enter the following information:

Field Element	Value
What do you want to call your integration?	ERP PO Event VBCS <ClassId> <StudentId> Ex: ERP PO Event VBCS 96 06
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of 01.00.0000. Alternatively, if you want to change the version number, enter the version using numbers only in this format: xx.xx.xxxx.
What does this integration do?	This integration demonstrates the use of Oracle Integration ERP Cloud Adapter Eventing capability along with the REST adapter to store the PO record in VBCS PO Table
Which package does this integration belong to?	Leave blank

Create Integration - Select a Style

How would you like to build your integration? Select a style to use.

App Driven Orchestration  <p>Multi-step Integration flow triggered by an Application or API.</p> <p>Select</p>	Scheduled Orchestration  <p>Multi-step Integration flow triggered by a Schedule. Commonly used for Batch/Bulk Integrations or File processing.</p> <p>Select</p>	File Transfer  <p>Seamlessly and securely move files across the network.</p> <p>Select</p>
Basic Routing  <p>Basic App to App Routing with Data Mapping.</p> <p>Select</p>	Publish To OIC  <p>Publish messages from Apps to OIC Pub/Sub Channel.</p> <p>Select</p>	Subscribe To OIC  <p>Subscribe to messages from OIC Pub/Sub Channel.</p> <p>Select</p>

Cancel

Create New Integration

?



Create New Integration

Enter information that describes this integration.

Describe this integration Use a meaningful name and description that will help others find and understand this integration. The Identifier and Version can be set only when the integration is created. The combination of Identifier and Version must be unique.

* What do you want to call your integration?

ERP PO Event VBCS 96 06

* Identifier

ERP_PO_EVENT_VBCS_96_06

* Version

01.00.0000

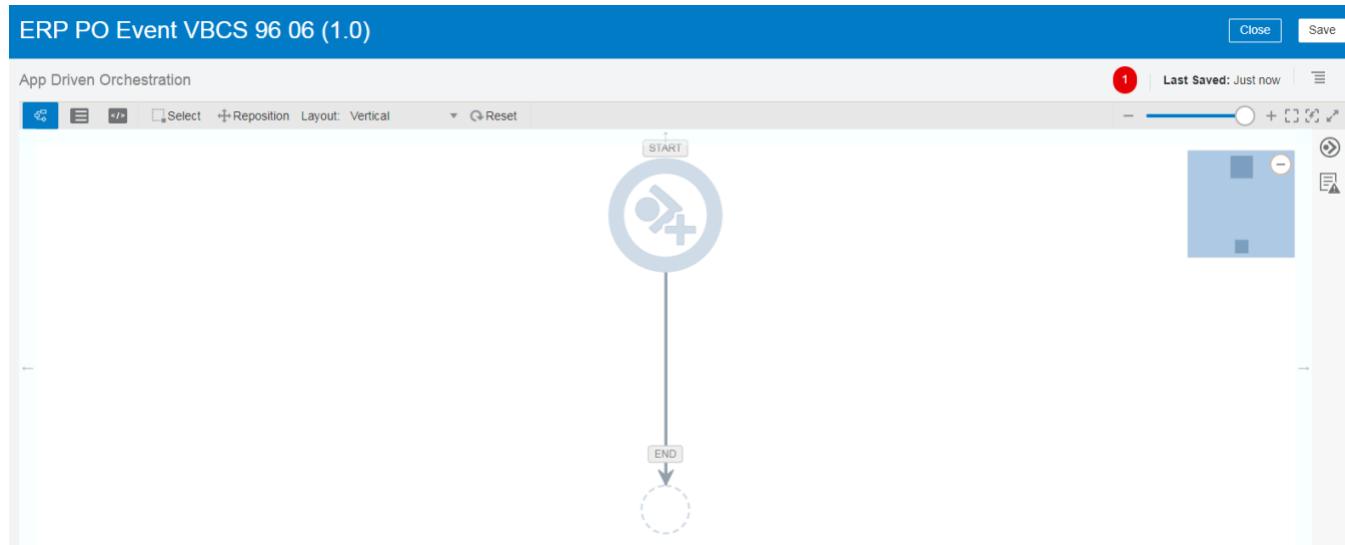
What does this integration do?

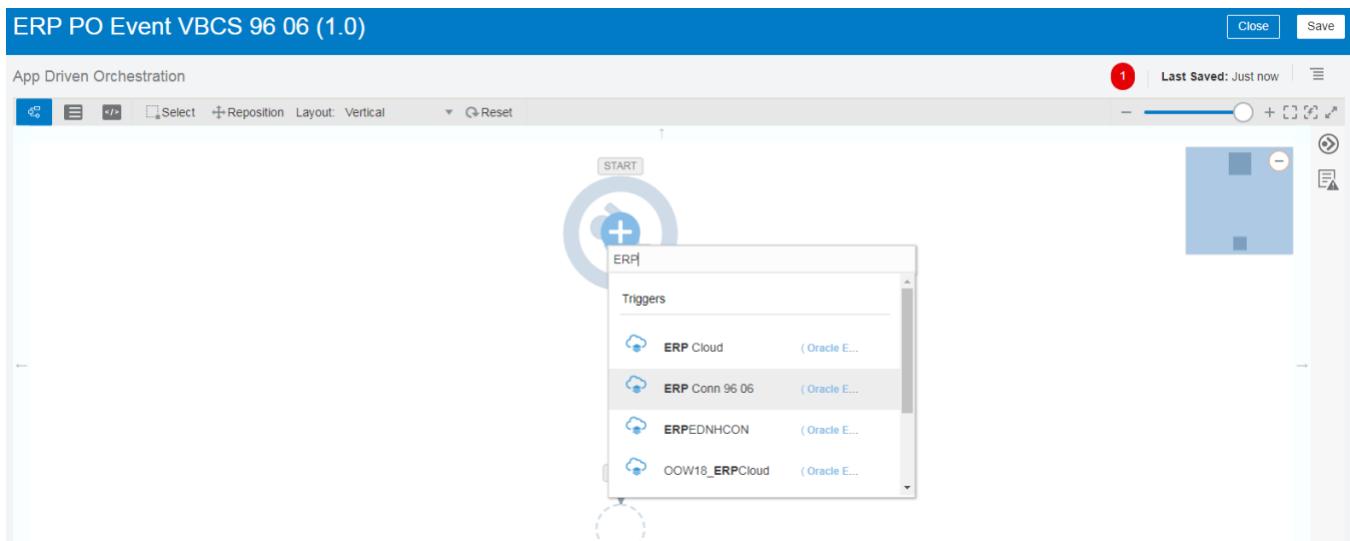
This integration demonstrates the use of Oracle Integration ERP Cloud Adapter Eventing capability along with the REST adapter to store the PO record in VBCS PO Table

Which package does this integration belong to?

Enter a new or existing package name

5. Click **Create**. The integration canvas is displayed.





6. Hover over the circle next to Start, and click on the + sign
7. Select the ERP Connection (ERP conn 96 06) from the list which is pre created for you
8. Enter details as in the screenshot below to define the endpoint in the flow

Welcome to the Oracle ERP Cloud Endpoint Configuration Wizard

This wizard helps you configure an endpoint using the Oracle ERP Cloud connection. You will be asked to specify configuration parameters and define an operation for the service.

Basic Info

* What do you want to call your endpoint?
POEVENT

What does this endpoint do?
This will subscribe to the PO Events

9. In the next screen, select “Receive Business Events raised within ERP Cloud” as the option and search for the “Purchase Order Event” and provide an XPath Expression as below and Click Next

```
<xpathExpr xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/">
<ns0:purchaseOrderLine><ns0:itemDescription>Lan Cable</ns0:itemDescription></ns0:purchaseOrderLine>
</xpathExpr>
```

Note: Change the highlighted value to some Item Description of your choice to filter the PO Record that you would be creating in ERP Cloud later. This is important as to filter your PO because multiple Users would be creating PO Event flows.

Configure Oracle ERP Cloud Endpoint

Help ▾ Back Next Cancel Done



Configure the Integration Service Endpoint to Receive Requests from the Oracle ERP Cloud Application

Select the business object or event that you want to receive from the Oracle ERP Cloud application as a request document to start this integration flow.

 Basic Info**Request**

Response

Summary

Configure a Request

 With Business Objects With Business Events Receive Status of ERP Import Job

* Business Event For Subscription

Purchase

Filter Expr for Purchase Order Event

Purchase Order Event

```
<xpathExpr
xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2"
xmlns:ns2="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/types/">$eventPayload/ns2:result/ns0:Value/ns0:PurchaseOrderLine/ns0:itemDescription/ns0:Lan Cable"></xpathExpr>
```

Event Description

The public business event is published when a purchase order is created, changed, finally closed, or reopened.

10. Select the Response type as “None” and Click Next

Configure Oracle ERP Cloud Endpoint

Help ▾ Back Next Cancel Done



Configure the Response Type to Send to the Oracle ERP Cloud Application

Select the business object to send as a response document from the integration flow to the Oracle ERP application. The immediate response option provides a request and response message exchange. The delayed response option provides an asynchronous response message exchange.

 Basic Info Request**Response**

Summary

Response Type

 Delayed None

Select the operation and business objects to configure for a successful callback response. You can also optionally select to configure a callback response for a failed integration flow.

You have selected to send no response document back to the source application. No further configuration is required.
Select Next to go to the Summary page.

11. Review summary page and Click Done

Configure Oracle ERP Cloud Endpoint

Help < Back Next > Cancel Done

Oracle Enterprise Resource Planning Business Event Subscription Summary
Oracle ERP Cloud endpoint configuration was successful.

POEVENT

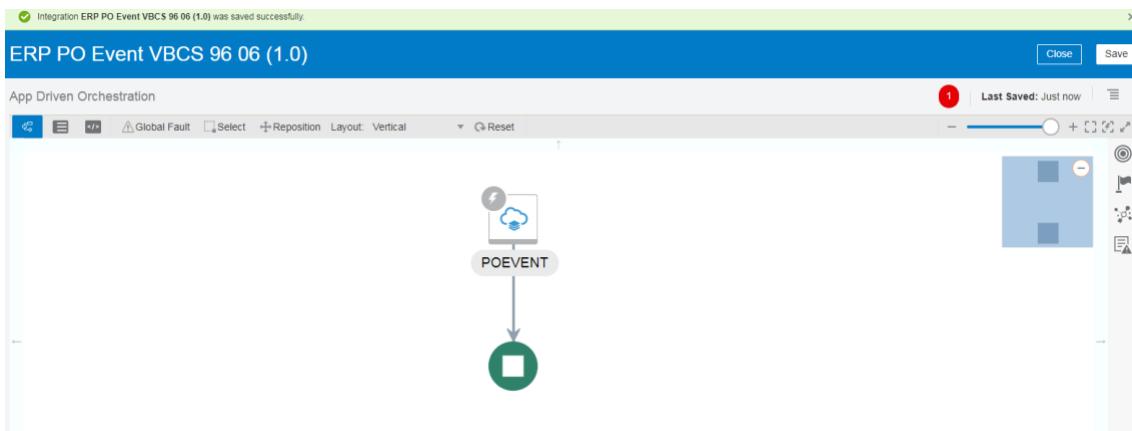
Description
This will subscribe to the PO Events

Subscribed Event: Purchase Order Event

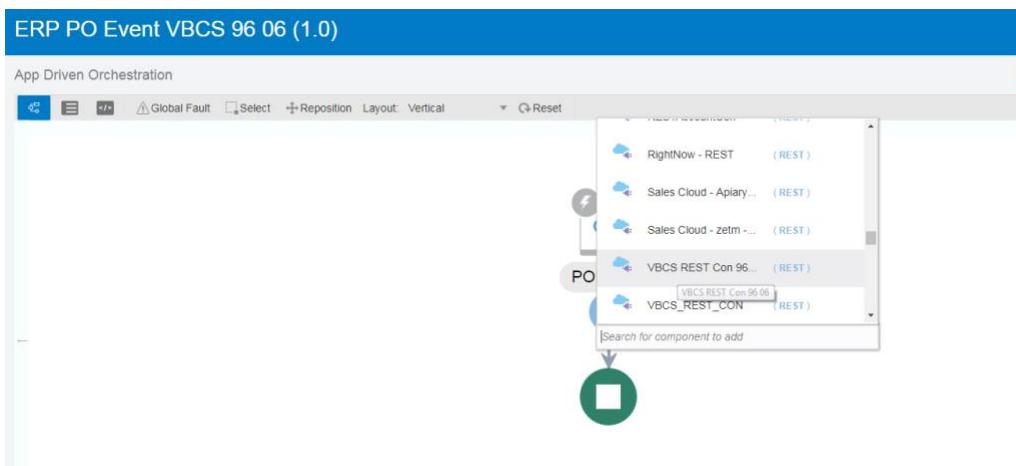
Filter Expression: <xpathExpr xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/" xmlns:ns1="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/">/ns0:purchaseOrder/ns1:header/ns1:orderNumber</xpathExpr>

Basic Info Request Response Summary

12. Integration so far



13. Next we will add the Invoke Action Using the REST connection created earlier to verify if the PO exists in VBCS PO table.
Hover on the wiring next to the “POEVENT” and click on “+” sign and Select the REST Connection (VBCS_REST_CON_<ClassId>_<StudentId>)



14. In the “Basic Info” page provide the details as per the screenshot below and Click Next

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the REST adapter.

Basic Info

* What do you want to call your endpoint?
GETPO

What does this endpoint do?
Gets PO information from VBCS if exists

* What is the endpoint's relative resource URI?
/PO

* What action do you want to perform on the endpoint?
GET

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

Add and review parameters for this endpoint

Configure a request payload for this endpoint

Configure this endpoint to receive the response

Configure Request Headers? Standard Custom

15. In the Request Parameters page Add a query parameter as per below and Click Next. We wanted to search for a PO which can be invoked using below query parameter format

Ex:

`https://<oid_host>/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO?q=pOHeaderId=300000074157561`

Configure REST Endpoint

Configure the Request Query Parameters
Configure the request query parameters for this endpoint.

Request Parameters

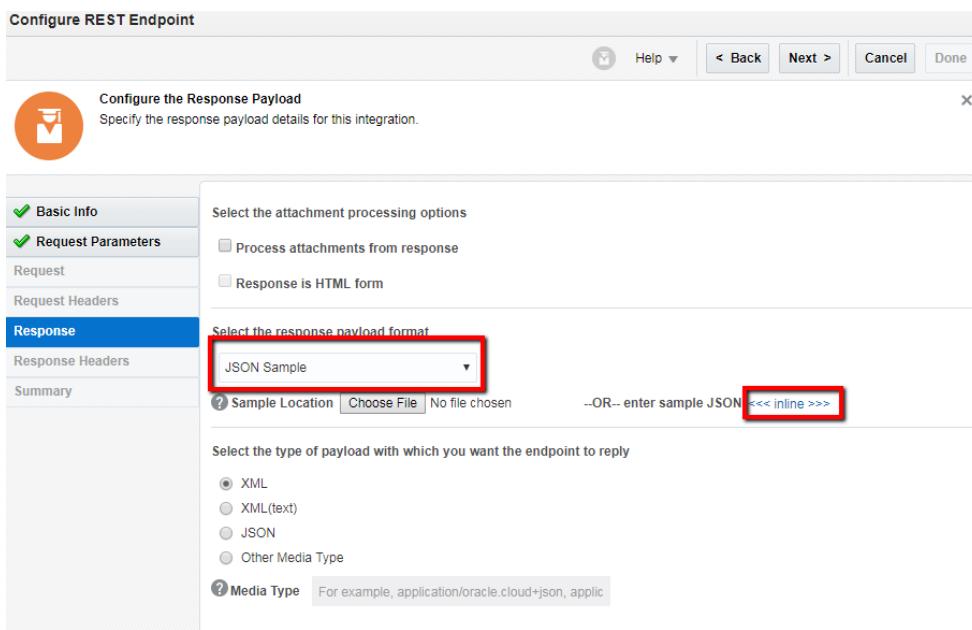
* Resource URI **/PO**

Specify Query Parameters

Name	Data Type
q	string

Template Parameters
Displays the template parameters in the relative resource URI. Template parameters are determined by details you specified (

16. In the Response page Select the Response payload format as JSON



Select "inline" and enter the below JSON sample in the text area and Click OK. Select Next

```
{
  "items": [
    {
      "id": 24,
      "creationDate": "2019-01-13T23:12:09+00:00",
      "lastUpdateDate": "2019-01-13T23:12:09.027+00:00",
      "createdBy": "john.doe@example.com",
      "lastUpdatedBy": "john.doe@example.com",
      "pOHeaderId": "300000074157561",
      "orderNumber": "162180",
      "procurementBUI": "300000046987012",
      "procurementBusinessUnit": "US1 Business Unit",
      "supplierId": "300000047414679",
      "supplier": "Dell Inc.",
      "soldToLegalEntity": "US1 Legal Entity",
      "IOCID": null,
      "links": [
        {
          "rel": "self",
          "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
          "name": "PO",
          "kind": "item",
          "properties": {
            "changeIndicator": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787
000000001770400000001737200106A6176612E6C616E672E446F75626C6580B3C24A296BFB0402000144000576616C
7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B02000078703FF00000000000000078"
          }
        },
        {
          "rel": "canonical",
          "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
        }
      ]
    }
  ]
}
```

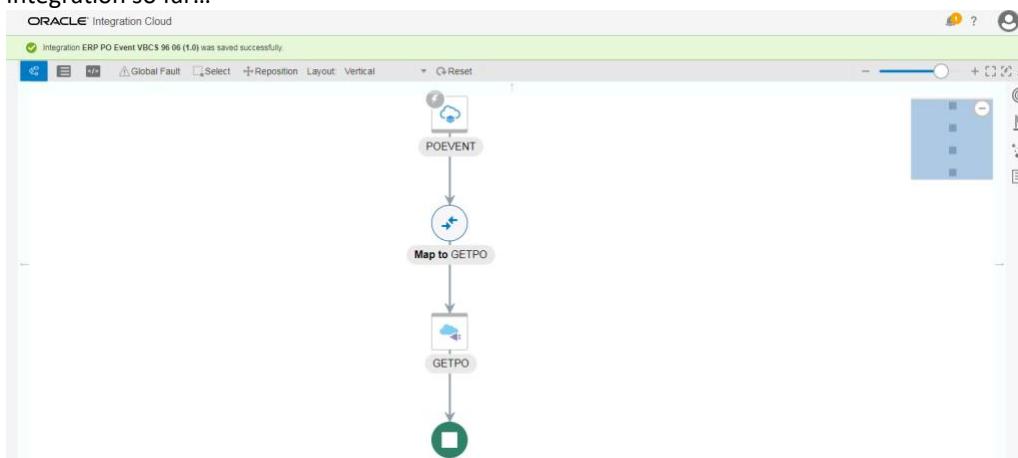
```

        "name": "PO",
        "kind": "item"
    }
]
}
],
"count": 1,
"hasMore": false,
"limit": 25,
"offset": 0,
"links": [
{
    "rel": "self",
    "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO",
    "name": "PO",
    "kind": "collection"
}
]
}
}

```

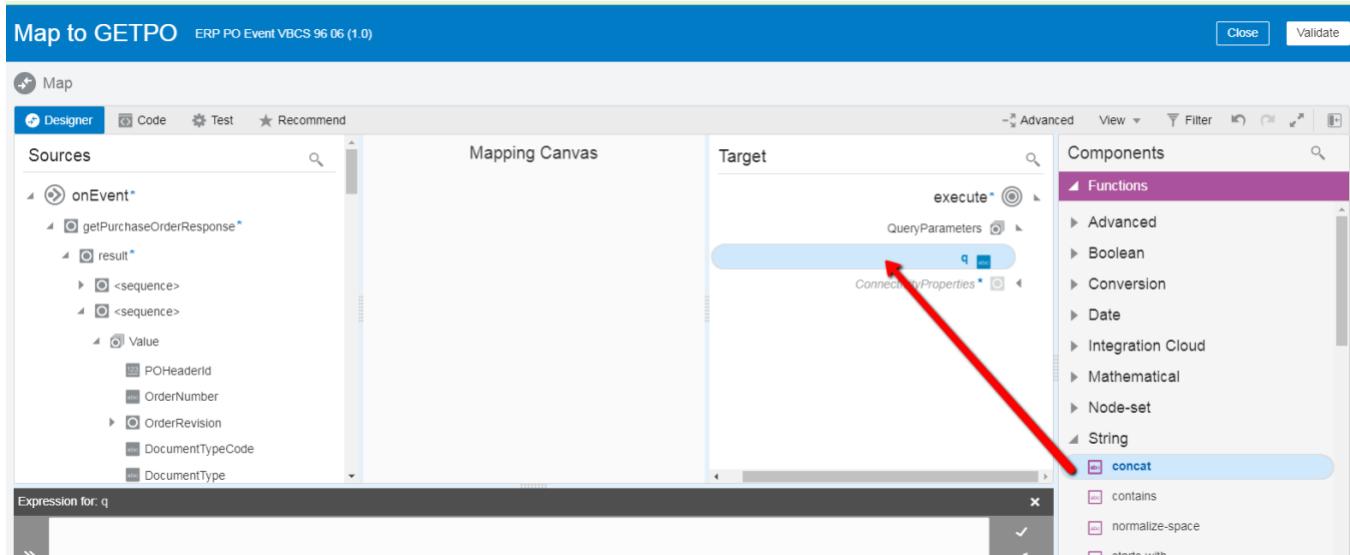
17. In the Summary page review the information and click Done

18. Integration so far...



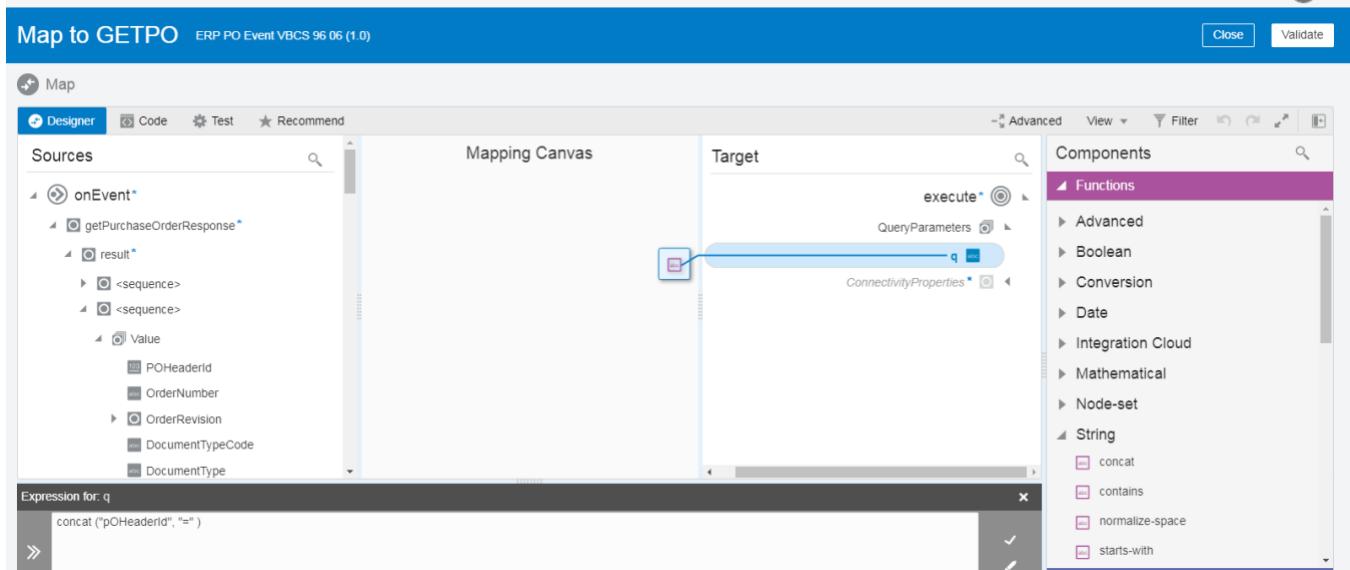
19. A Map Action (Map to GETPO) is created automatically. Perform the mapping as per below. Select the Map Action and Click on "Edit" to open the Mapper

Select the "q" Query Parameter from the Target, drag and drop the concat function to build query parameter value as "pOHeaderId=300000074157561".



Provide static value "pOHeaderId" as the first parameter value for the concat("", "", "")

Provide static value "=" as the second parameter value for the concat("", "", "")



Provide the third parameter value by dragging the "POHeaderId" element inside the concat("", "", "")

The screenshot shows the Oracle Integration Map to GETPO Designer interface. On the left, the Sources pane lists an *onEvent** source containing a *getPurchaseOrderResponse** event with a *result** sequence. The Mapping Canvas shows a flow from this source to a Target section. In the Target section, there is an *execute** component with a blue connector labeled 'q'. Below the canvas is an Expression window with the following text:

```
concat ("poHeaderId", "=")
```

A red arrow points from the text in the Expression window to the window itself.

Click on the Tick mark in the expression window to save your expression

The screenshot shows the Oracle Integration Map to GETPO Designer interface. The expression in the Expression window has been updated to:

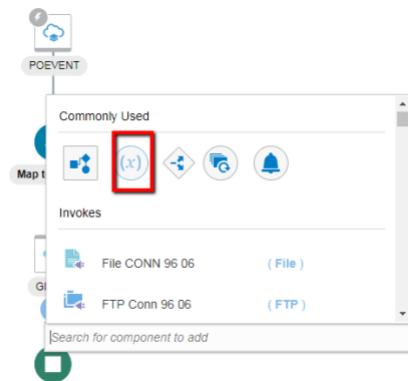
```
concat("poHeaderId", "=") /nssrcmpr:onEvent/inp1:getPurchaseOrderResponse/inp1:result/ns4:Value/ns4:POHeaderId
```

Click on Validate to check for any errors.

The screenshot shows the Oracle Integration Map to GETPO Designer interface. At the top, a green bar displays the message: "Mapping is valid and ready to use." The main interface shows the same mapping configuration as the previous screenshots, with the expression now fully defined in the Expression window.

Save your integration

20. Now, we will assign PO Record Count value to a variable “POHeaderId”
 Hover on the wiring next to “GETPO” and click on “+” sign and Select Assign Action



Provide a name for the Assign Action and Click on Create

Create Action

Assign

Please give a unique name and description to this action.

* Name	PORecord
Description	Enter description

Create **Cancel**

Click on “+” to add a new Variable. Add a variable “PORecordCount” of Data type string

PORecord ERP PO Event VBCS 96 06 (1.0) **Close** **Validate**

(x) Assign

Assign variables to your integration. You can assign values to variables using the editor. Variable assignments can be a greater of complexity. For example, you can use assignments in other activities and in maps.

Add at least one named variable and specify its value by adding an expression.

Variable	Data Type	Description	Operation	Value
(x) PORecordCount	string	Type a description		Add an expression

+

Click the pencil icon to “Add an expression” to assign record count value. In the expression builder drag and drop the count element that we got as a response from the GETPO API call from the earlier step.

PORecordCount

Expression in "PORecord - ERP PO Event VBCS 96 06 (1.0)"

Inputs

Source: Find...

- onEvent
 - *getPurchaseOrderResponse
 - *result
- SGETPO
 - *executeResponse
 - *response-wrapper
 - *Items
- *count

View Detach

Components

Find...

- Functions
- Operators

Expression

```
$GETPO/nssrcmpr:executeResponse/nssrcdfi:response-wrapper/nssrcdfi:count
```

Expression Summary

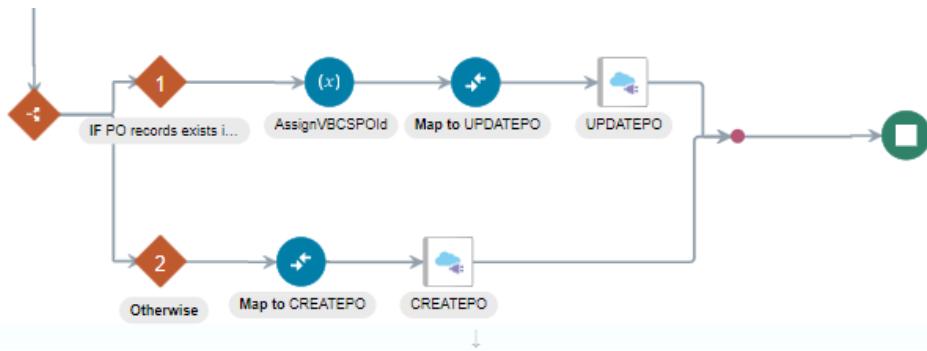
Validate and Close the Expression Builder...Save your integration

Integration so far...

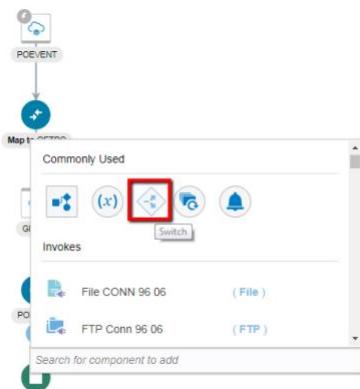


21. Next we will try to build an if-else logic as per below pseudo code

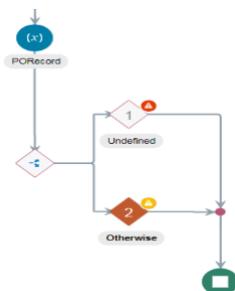
Search for PO record based on POHeaderID (Implemented already in previous steps)
 If (count of PORecords in VBCS) = 1
 - Update existing VBCS PO Record
 Else
 - Insert a new PO Record



Hover on the wiring next to the PORecord Activity and Select the “Switch” Action



Below construct is created



If – Flow (Condition 1)

Select Condition “1” and click on Edit icon which brings up an Expression builder and provide the condition as per below

Expression Name	
PO records exists in VBCS	
<input type="button" value="Close"/> <input type="button" value="Validate"/> <input type="button" value="Expression Mode"/>	
Expression in "ERP PO Event VBCS 96 06 (1.0)"	
Inputs	
View <input type="button" value="Filter"/> <input type="button" value="Detach"/>	
Source <input type="text" value="Find..."/> <input type="button" value="Find"/>	
> < *result & <> \$GETPO & <> *executeResponse > < *response-wrapper \$PORecordCount \$Tracking_var_1	
Components	
View <input type="button" value="Filter"/> <input type="button" value="Detach"/>	
Find... <input type="button" value="Find"/>	
Functions Operators	
Expression Name PO records exists in VBCS	
New Condition >> \$PORecordCount = "1"	

22. Hover on the wiring next to the If Condition (1) Action and Click on “+” Sign and select the “Assign” Action

Provide a name for the Assign Action as “AssignVBCSPOId”

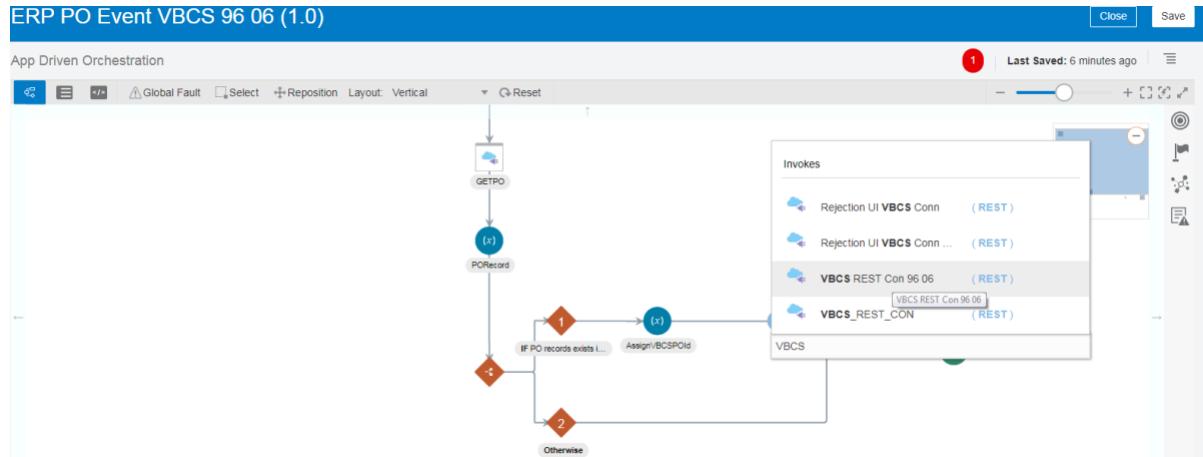
Create a Variable “VBCSPOId” of String data type and add an expression as below. Validate and Close the expression builder.

The screenshot shows the Expression Builder interface for the 'AssignVBCSPOId' action. The 'Expression' field contains the path '\$GETPO/nssrcmpr.executeResponse/nssrcdff:response-wrapper/nssrcdff:items/nssrcdff:id'. The 'Source' tree on the left shows the structure of the response message, with a red arrow pointing to the 'id' node under the 'items' node. The status bar at the bottom indicates that the assign action is valid and ready to use.

The screenshot shows the 'AssignVBCSPOId' configuration window. It displays a table with one row. The table has columns for 'Variable', 'Data Type', 'Description', 'Operation', and 'Value'. The 'Variable' column contains '(x) VBCSPOId', 'Data Type' is 'string', 'Description' is 'Type a description', 'Operation' is a pencil icon, and 'Value' is 'id'. The status bar at the bottom indicates that the assign action is valid and ready to use.

Validate and Close the Assign window. Save your integration.

23. Hover on the wiring next to the Assign Action and Click on “+” Sign and select the REST Connection created earlier ex: VBCS_REST_CON_<ClassId>_<StudentId>



24. In the Basic Info Page provide the details as per below and Click Next

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the REST adapter.

Basic Info

Request Parameters	* What do you want to call your endpoint? UPDATEPO	
Request	What does this endpoint do? updates po with vbc cross reference id	
Request Headers	* What is the endpoint's relative resource URI? /PO/{ID}	
Response	* What action do you want to perform on the endpoint? PATCH	
Response Headers	Based on your selections, you can add parameters or configure a request and/or response for this endpoint.	
Summary	Select any options that you want to configure:	
	<input checked="" type="checkbox"/> Add and review parameters for this endpoint <input checked="" type="checkbox"/> Configure a request payload for this endpoint <input checked="" type="checkbox"/> Configure this endpoint to receive the response	

25. Nothing to Specify in Review Parameters page. Click Next

The screenshot shows the 'Configure REST Endpoint' dialog box. On the left, a sidebar lists 'Basic Info', 'Request Parameters' (which is selected), 'Request Headers', 'Response Headers', and 'Summary'. The main area is titled 'Configure the Request Query Parameters' and contains a sub-section 'Specify Query Parameters' with a table for adding parameters. A note below says 'Template Parameters'.

26. In the Request Page Select the request payload format as "JSON Sample". Select the "inline" link and provide the below Sample JSON request and Click Next

The screenshot shows the 'Configure REST Endpoint' dialog box. The sidebar shows 'Basic Info', 'Request Parameters', and 'Request' (selected). Under 'Request', 'Request Headers' and 'Response Headers' are listed. The main area shows 'Select the attachment processing options' with checkboxes for 'Send attachments in request' and 'Request is HTML form'. It also shows 'Select the request payload format' with a dropdown set to 'JSON Sample', a 'Sample Location' section, and an 'Element' dropdown set to 'request-wrapper'. Below this, 'Select the type of payload with which you want the endpoint to send' has 'JSON' selected. A 'Media Type' input field is at the bottom.

```
{
  "orderNumber": "163521",
  "procurementBuild": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell.",
  "soldToLegalEntity": "US1 Legal Entity",
  "IOCID": 2
}
```

27. In the Response page Select the request payload format as "JSON Sample". Select the "inline" link and provide the below Sample JSON Response and Click Next

```
{
  "id": 24,
  "creationDate": "2019-01-13T23:12:09+00:00",
  "lastUpdateDate": "2019-01-22T16:46:08+00:00",
  "createdBy": "john.doe@example.com",
  "lastUpdatedBy": "john.doe@example.com",
  "pOHeaderId": "300000074157561",
  "orderNumber": "163521",
  "procurementBUIlD": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell.",
  "soldToLegalEntity": "300000046973970",
  "IOClId": null,
  "links": [
    {
      "rel": "self",
      "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
      "name": "PO",
      "kind": "item",
      "properties": {
        "changeIndicator": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787
0000000017704000000001737200106A6176612E6C616E672E446F75626C6580B3C24A296BFB0402000144000576616C
7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B020000787040000000000000000078"
      }
    },
    {
      "rel": "canonical",
      "href": "https://oic_host/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/24",
      "name": "PO",
      "kind": "item"
    }
  ]
}
```

28. Review the Summary Page and Click on Done

Configure REST Endpoint

REST Endpoint Configuration Summary

REST endpoint configuration was successful.

Basic Info
Request Parameters
Request
Request Headers
Response
Response Headers
Summary

Rest endpoint summary

UPDATEPO

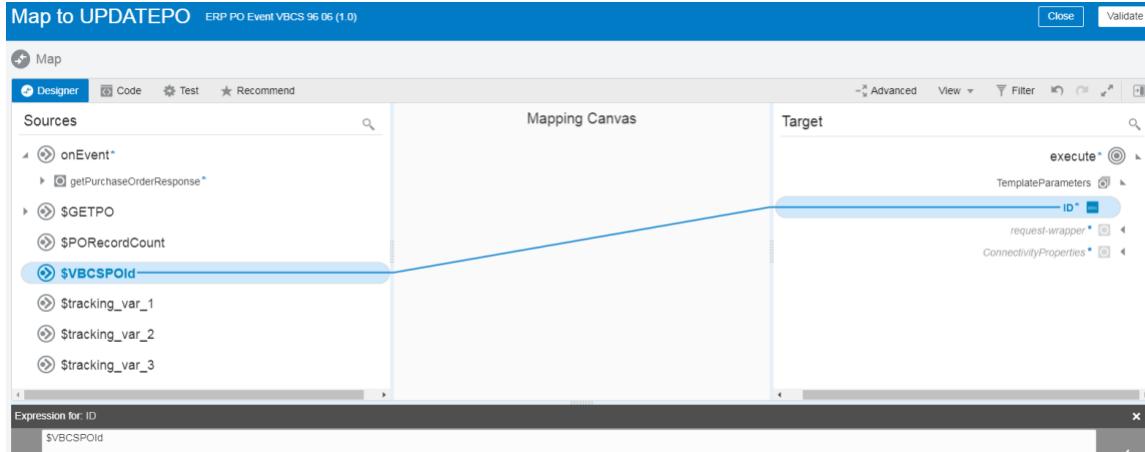
Description: updates po with vbcS cross reference id

Endpoint Summary

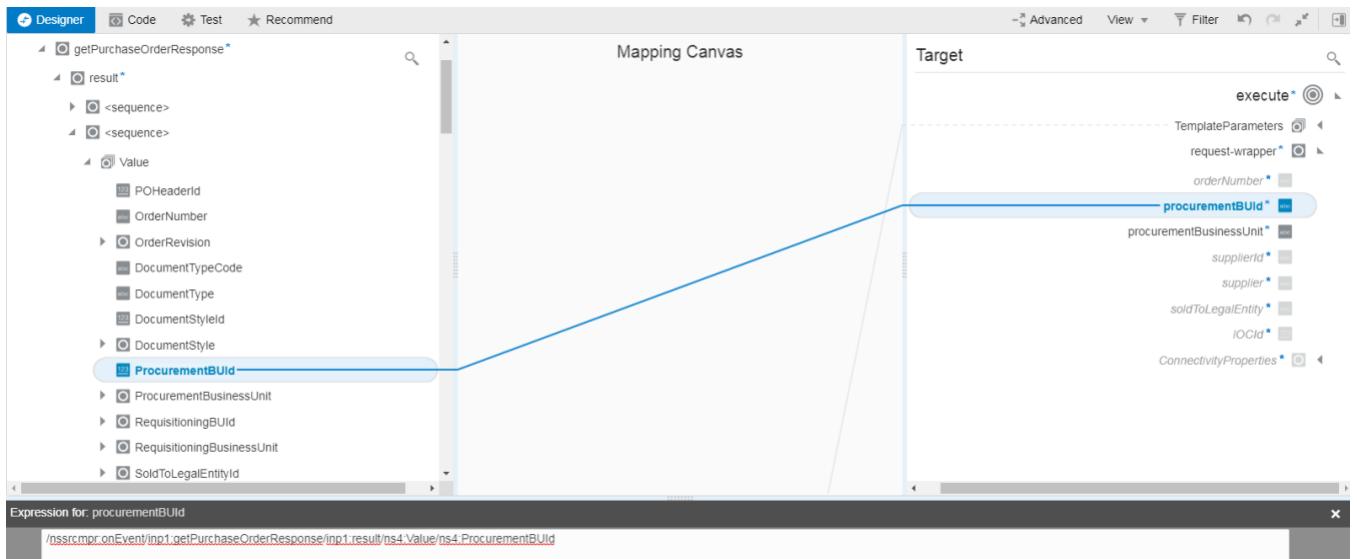
- REST Service URI: /PO/{ID}
- Method: PATCH
- Request Media Type: JSON
- Response Media Type: JSON

29. A Map Action (Map to UPDATEPO) is created. Select the Map Action and Click on Edit Icon and provide mapping as per below screenshots

a. Map \$VBCSPOId TO TemplateParameters-> ID



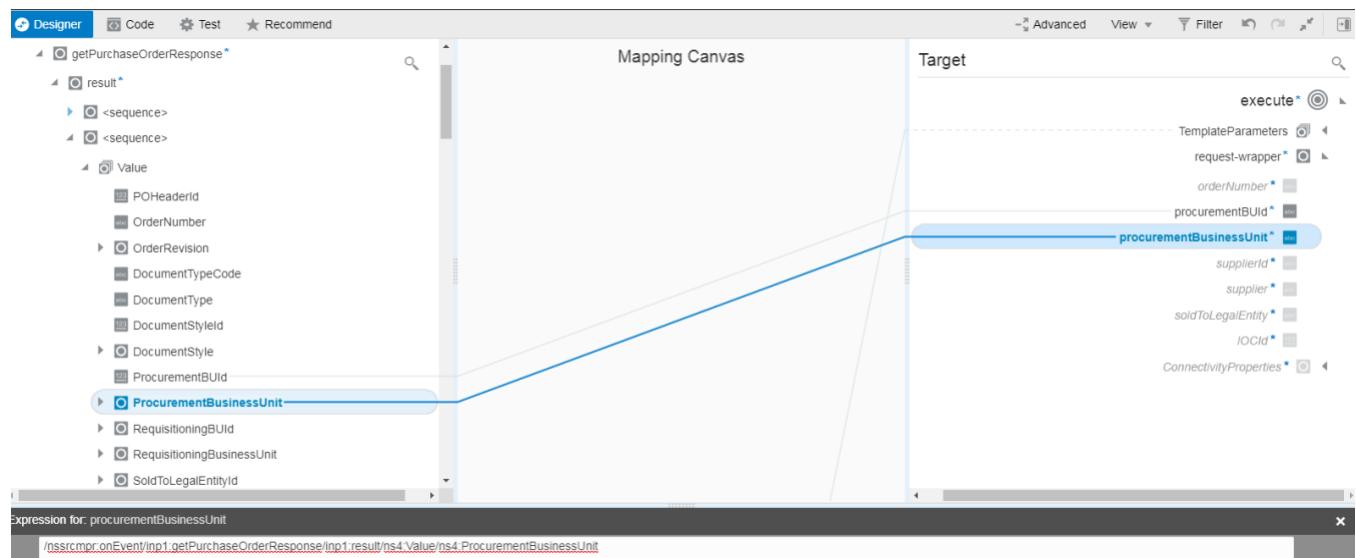
b. Map onEvent->getPurchaseOrderResponse->result-><sequence>->Value-> ProcurementBUId
TO request-wrapper -> procurementBUId



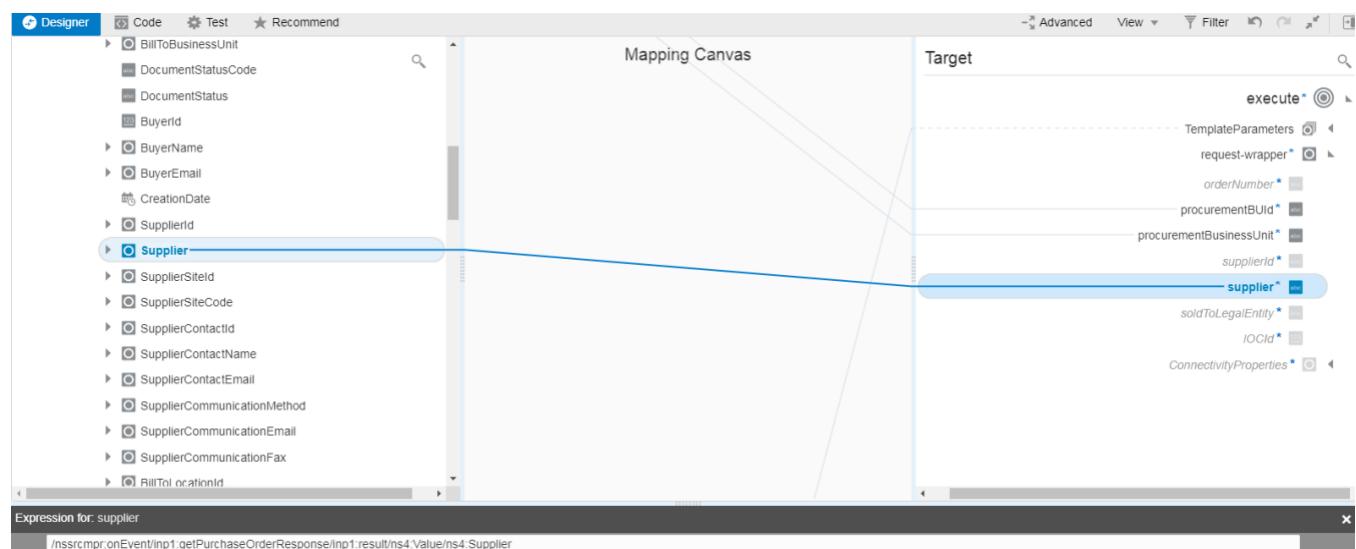
c.
d.

Oracle Product Management

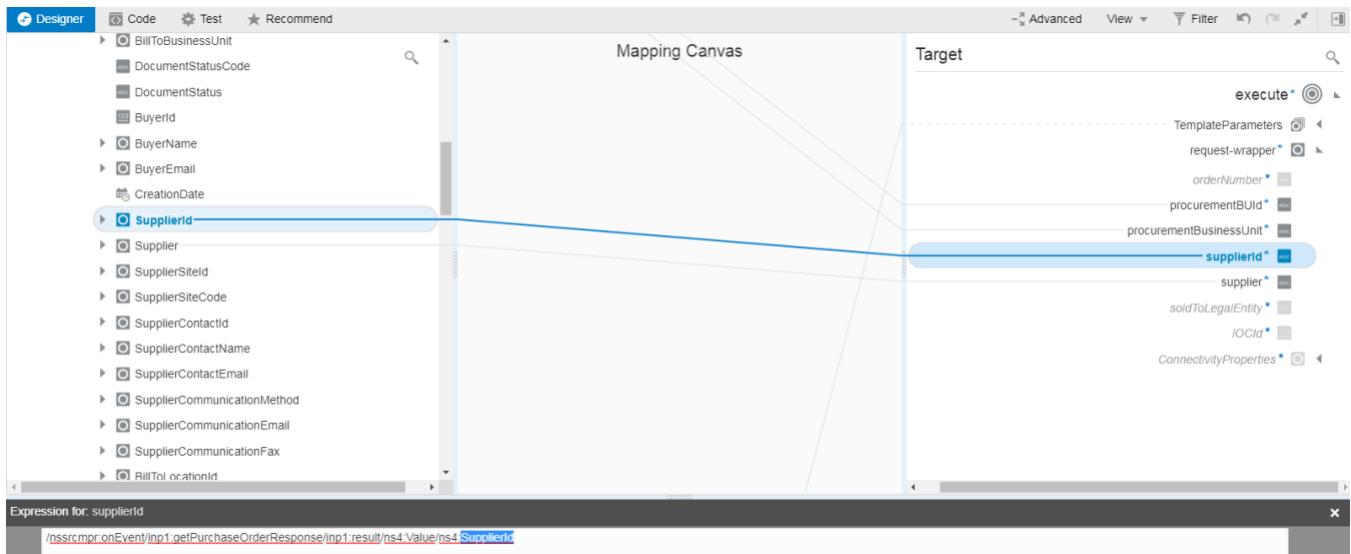
- e. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> ProcurementBusinessUnit
TO request-wrapper -> procurementBusinessUnit



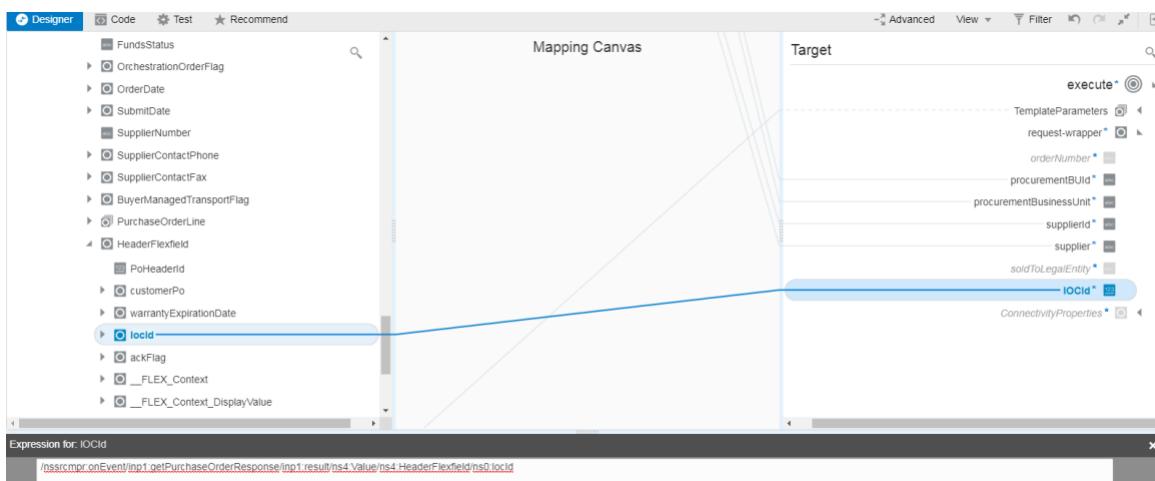
- f. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> Supplier
TO request-wrapper -> supplier



- g. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> SupplierId
TO request-wrapper -> supplierId

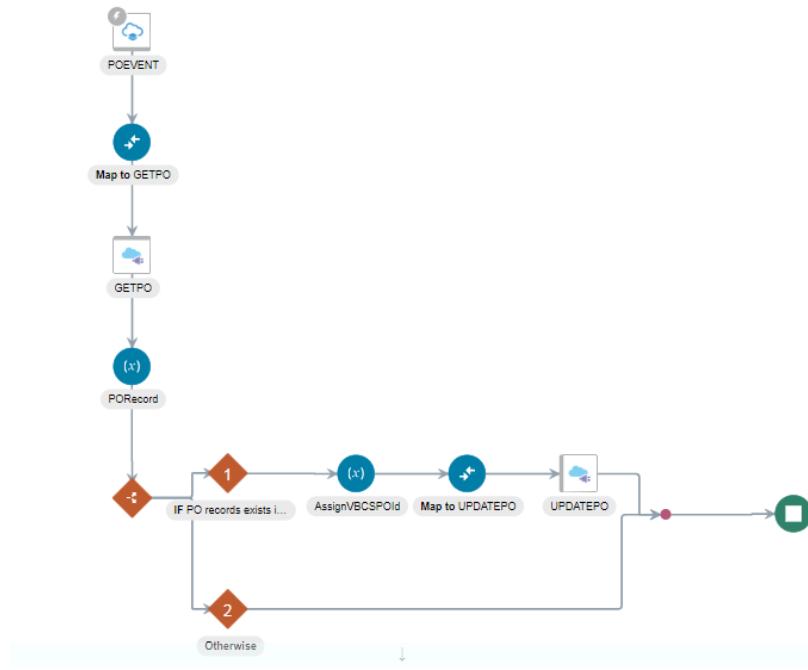


- h. Map onEvent -> getPurchaseOrderResponse->result-><sequence>->Value-> HeaderFlexfield->locId
TO request-wrapper -> IOClD



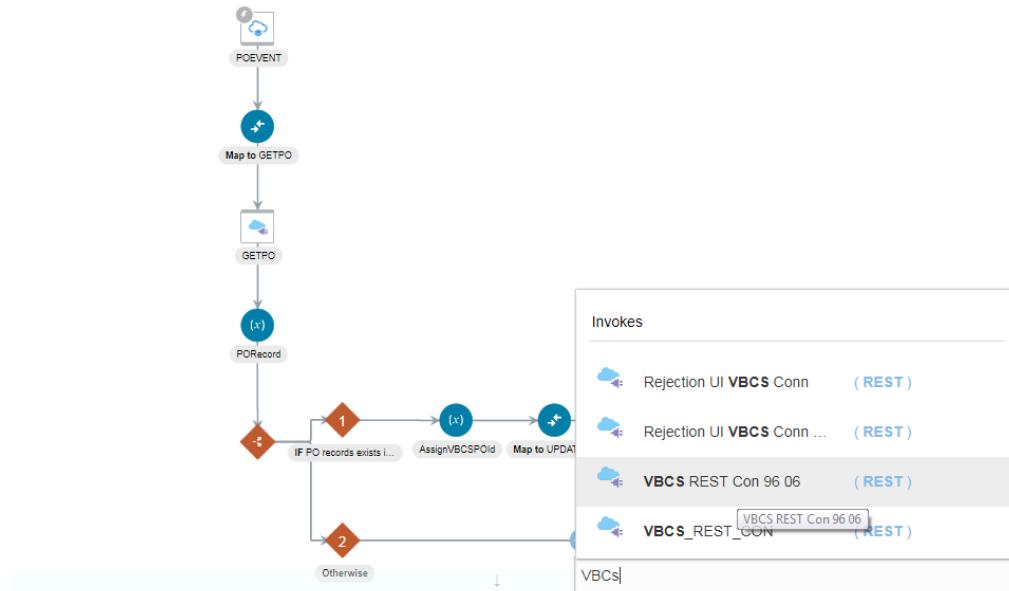
Click on Save and Validate. Close the mapper

30. Integration flow so far...

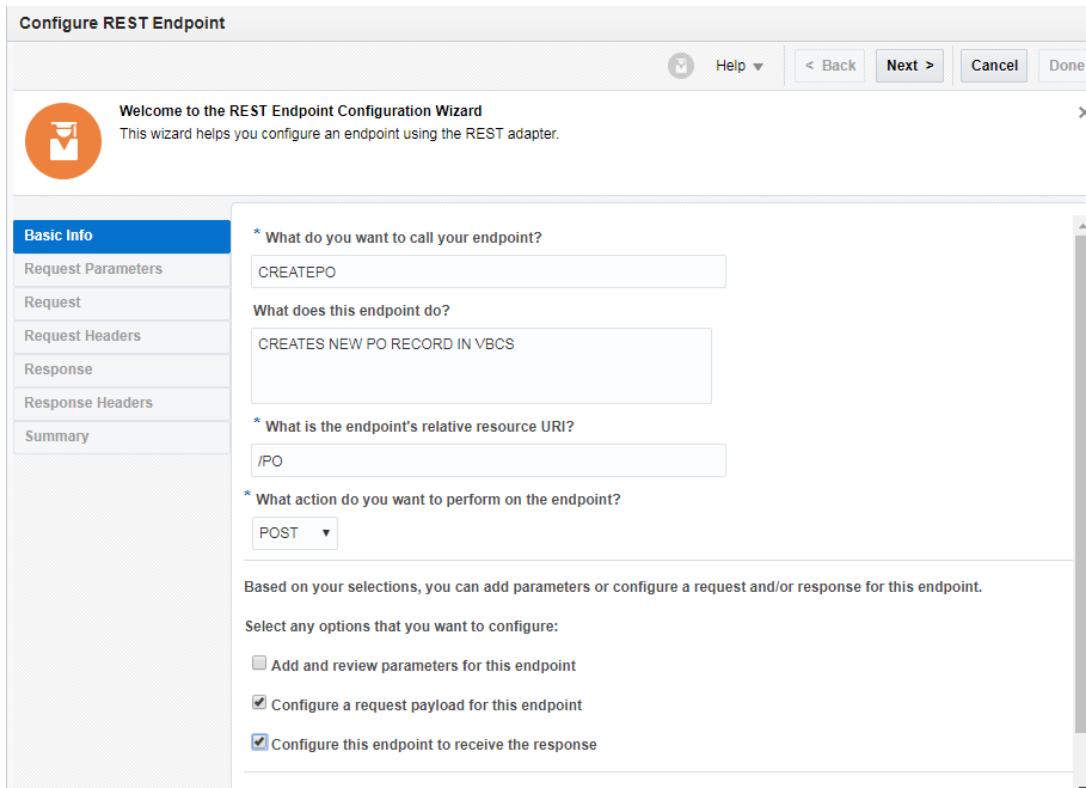


31. Else – Flow

Hover on the wiring next to “Otherwise” Action and Select VBCS_REST_CON_<ClassId>_<StudentId> connection



32. In Basic Page configure the details as per below



33. In the Request Page Select the request payload format as “JSON Sample”. Select the “inline” link and provide the below Sample JSON request and Click Next

```
{
  "pOHeaderId": "300000074157551",
  "orderNumber": "163521",
  "procurementBUId": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell Inc.",
  "soldToLegalEntity": "US1 Legal Entity"
}
```

34. In the Response Page Select the response payload format as “JSON Sample”. Select the “inline” link and provide the below Sample JSON response and Click Next

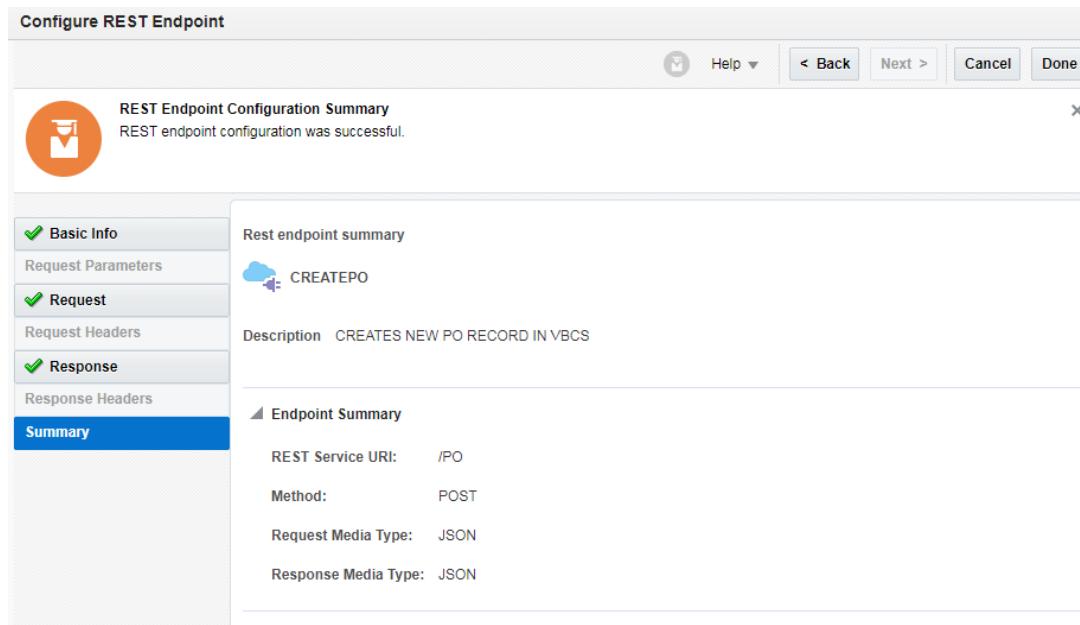
```
{
  "id": 41,
  "creationDate": "2019-01-22T17:32:24+00:00",
  "lastUpdateDate": "2019-01-22T17:32:24.027+00:00",
  "createdBy": "john.doe@example.com",
  "lastUpdatedBy": "john.doe@example.com",
  "pOHeaderId": "300000074157551",
  "orderNumber": "163521",
  "procurementBUId": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "supplierId": "300000047414679",
  "supplier": "Dell Inc.",
  "soldToLegalEntity": "300000046973970",
  "IOCID": null,
```

```

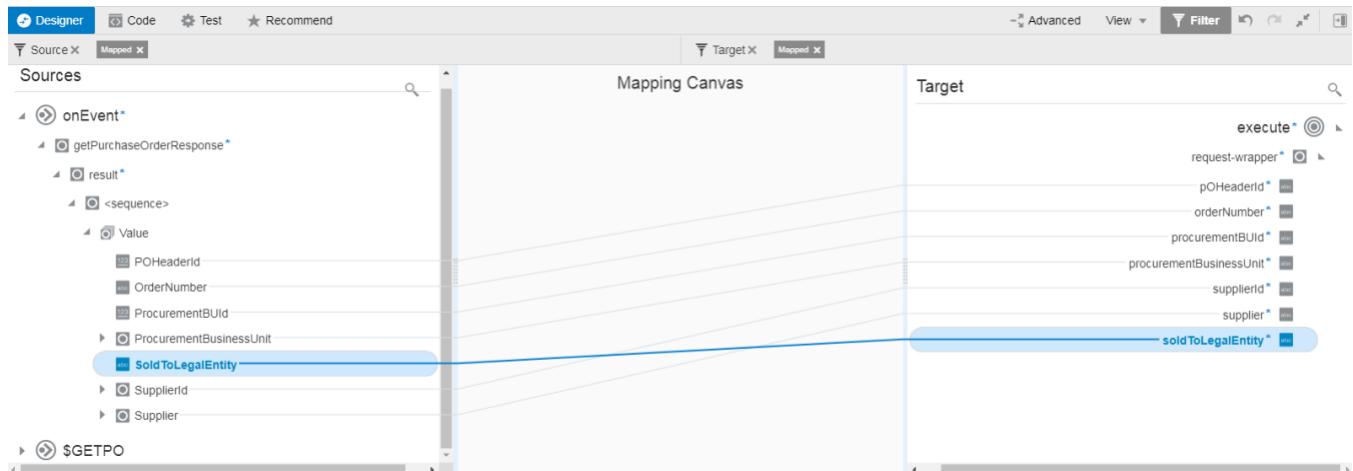
"links": [
  {
    "rel": "self",
    "href": "https://oichost/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/41",
    "name": "PO",
    "kind": "item",
    "properties": {
      "changeIndicator": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A657870000
00001770400000001737200106A6176612E6C616E672E446F75626C6580B3C24A296BFB0402000144000576616C75657872
00106A6176612E6C616E672E4E756D62657286AC951D0B94E08B02000078703FF0000000000000078"
    }
  },
  {
    "rel": "canonical",
    "href": "https://oichost/ic/builder/design/LetterOfCreditPortal/1.0/resources/data/PO/41",
    "name": "PO",
    "kind": "item"
  }
]
}

```

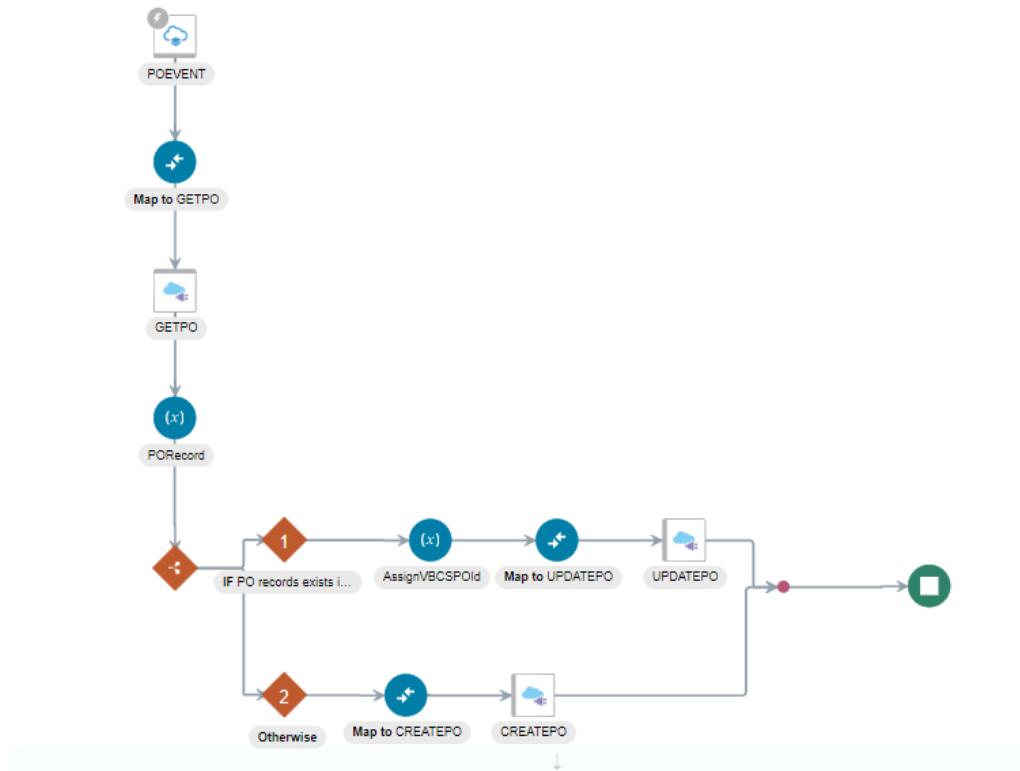
35. Review the Summary and Click Done



36. A Map Action is created. Provide the mapping for the Map Action (Map to CREATEPO) as per the screenshot below



37. Final Flow 1 Integration



Save your integration flow.

38. While you are in the Integration flow page, Select the Hamburger icon on the TOP right corner and click on Tracking. Configure OrderNumber, POHeaderId and ItemDescription as tracking identifiers to identify your instance quickly when we a PO is created in ERP Cloud. Click Save on the Tracking page.

The screenshot shows the Oracle Product Management interface for configuring business identifiers. On the left, there's a tree view under 'Source' showing a hierarchy of integration steps: 'onEvent' -> 'getPurchaseOrderResponse' -> 'result' -> 'Value' -> 'POHeaderId' -> 'OrderNumber'. To the right is a table for defining tracking fields:

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text
<input checked="" type="checkbox"/>	OrderNumber	Order Number	tracking_var_1	How to track it?
	POHeaderId	PO Header Id	tracking_var_2	How to track it?
	ItemDescription	Item Description	tracking_var_3	How to track it?

Activate Integration (Flow1)

1. On the Integrations page, click on the “ERP PO EVENT VBCS <ClassId><StudentId>” flow using the slider button

The screenshot shows the Oracle Integration Cloud Integrations page. It lists three integration flows:

- Invoice Bulk Import Callback Review 96 06 (1.0)**: APP DRIVEN ORCHESTRATION. Status: Enabled (indicated by a green switch).
- ERP PO Event VBCS 96 06 (1.0)**: APP DRIVEN ORCHESTRATION. Status: Enabled (indicated by a red box around the green switch).
- Invoice Bulk Import to ERP 96 06 (1.0)**: SCHEDULED ORCHESTRATION. Status: Enabled (indicated by a grey switch).

2. Check the *Oracle Recommends* check box to contribute your mappings to the Recommendations engine that then suggests mappings to you and other users for similar integrations. Also, check *Enable tracing* and *Enable payload* for debugging and troubleshooting (you would typically have the tracking and payload options disabled on production instances, but we enable them here for this lab).

Jan 25, 2019 12:37:04 PM UTC

Activate Integration

ERP PO Event VBCS 96 06 (1.0)

Oracle Recommends

Contribute integration mappings to Oracle Recommends.

Oracle Asserter: When Asserter recording is enabled, payloads will be captured and integration instances will be recorded. Recordings can be played later and maximum five recordings will be maintained for an integration.

Enable Asserter Recording

Tracing: When tracing is enabled, integration activity can be viewed in the Activity Stream.

Enable tracing

Include payload

When payload is included, sensitive information from the payload is written into log files, which can be downloaded and viewed. This may pose a security risk, and also slow down your system. Not recommended in a production environment.

Learn More

Activate Cancel

3. The activation should complete in a few seconds typically and show a green ribbon at the top.

ORACLE Integration Cloud

Integrations

Import Create

Jan 25, 2019 12:54:26 PM UTC

Search Last Updated

ERP PO Event VBCS 96 06 (1.0)

APP DRIVEN ORCHESTRATION

This integration demonstrates the use of Oracle Integration ERP Cloud Adapter Eventing capability along with the REST adapter to store the F

TRACE WITH PAYLOAD

Testing the ERP Event Flow Integration (Flow1)

1. Login into ERP Cloud as casey.brown with the credentials provided
2. Select Procurement tab and click on Purchase Orders

vision

Good evening, Sr. Jorge Monge!

Me My Team **Procurement** Product Management Payroll Risk Management Tools Financial Reporting Compliance >

APPS

Purchase Requisitions Purchase Agreements **Purchase Orders** My Receipts Negotiations Catalogs Suppliers

3. Select Tasks tab and Click on Create Order

The screenshot shows the Oracle Product Management interface with the 'vision' theme selected. The left side features a dashboard with various metrics: 'Orders Requiring Attention' (0 Failed Submission, 1 With Invoice Holds, 0 Rejected), 'Orders in Process' (No data to display), 'Recent Activity' (Change order submitted | 1/13/19, Order 163521 - Dell Inc.), 'Requisition Lines' (0 Requiring Action), and 'Incomplete Orders' (0 Orders, 2 Change C...). Below these are sections for 'Open Schedules' (a large blue circle) and 'Aging Requisition Lines' (Days Unprocessed: 0 to 3 Days, 4 to 7 Days, 8 to 14 Days, Over 14 Days). The right side contains a navigation menu with several categories: Requisitions, Orders, Agreements, Deliverables, Supply Base, and Administration. Under 'Orders', the 'CREATE ORDER' option is highlighted with a red box.

4. Enter Supplier information (Dell Inc.) and leave the rest to default. Click on Create

The screenshot shows the 'Create Order' dialog box. It includes fields for 'Style' (Purchase Order), 'Procurement BU' (US1 Business Unit), 'Requisitioning BU' (US1 Business Unit), 'Supplier' (Dell Inc., highlighted with a red box), 'Supplier Site' (Dell US1), 'Supplier Contact' (Gill, Noah), 'Default Ship-to Location' (Seattle), 'Sold-to Legal Entity' (US1 Legal Entity), 'Currency' (USD), and 'Buyer' (Brown, Casey). At the bottom are 'Create' and 'Cancel' buttons.

5. In the Purchase Order page go to the Order Lines Section and provide Item information as below

Oracle Product Management

Edit Document (Purchase Order): 163531 ★

Social Manage Approvals View PDF Actions Save Submit Cancel

The screenshot shows the Oracle Product Management interface for a Purchase Order. At the top, it displays the order number (163531), status (Incomplete), buyer (Brown, Casey), creation date (1/25/19), communication method (None), bill-to location (Seattle), default ship-to location (Seattle), total amount (5.00 USD), and procurement card information. Below this, the 'Terms' section includes fields for required acknowledgment (None), acknowledge within days, payment terms (Net 30), shipping method (UPS), freight terms (Buyer pays freight), FOB (Origin), and various checkboxes for payment and delivery terms. The 'Additional Information' section contains fields for customer PO, warranty expiration date, letter of credit ID, and supplier acknowledgment flag. The 'Lines' section shows a single line item for a 'Lan Cable' with category 'Computer Supplies', quantity '1', UOM 'EA', price '5.00', location 'Seattle', and requester information.

Note: For “Description” enter a Unique Value of your choice, which was provided in the xpath expression earlier when configuring the ERP adapter PO Event

Ref:

The screenshot shows the configuration of an integration service endpoint for receiving requests from the Oracle ERP Cloud application. It includes sections for basic info, request, response, and summary. The request configuration is highlighted, showing options for configuring a request, selecting a business event for subscription (set to 'Purchase'), and defining an XPath expression for the purchase order event. The XPath expression is defined as follows:

```

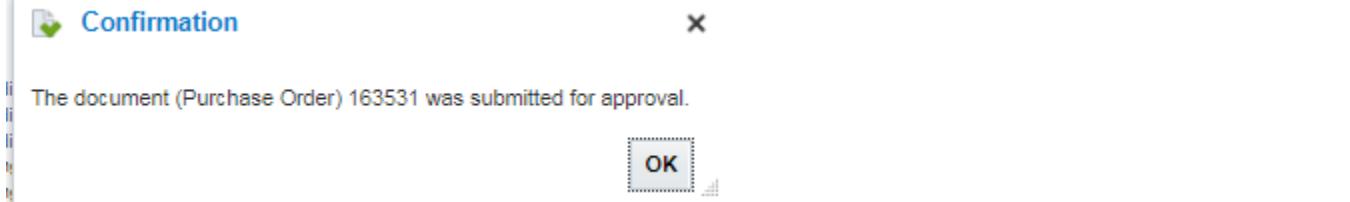
<xpathExpr
 xmlns:ns0="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2"
 xmlns:ns2="http://xmlns.oracle.com/apps/prc/po/editDocument/purchaseOrderServiceV2/types/">$eventPayload/ns2:result/ns0:value/ns0:PurchaseOrderLine/ns0:itemDescription/n= Lan Cable </xpathExpr>

```

The 'Event Description' section states: "The public business event is published when a purchase order is created, changed, finally closed, or reopened."

Oracle Product Management

6. Click on Submit to Create a Purchase Order. Make a note of the Order Number



7. Click on Tasks tab and Select Manage Order

The screenshot shows the Oracle Product Management interface with the 'Tasks' tab selected. On the left, there are several cards: 'Orders Requiring Attention' (0 Failed Submission, 1 With Invoice Holds, 0 Rejected), 'Orders in Process' (No data to display), 'Recent Activity' (listing recent orders and payments), 'Requisition Lines' (0 Requiring Action), and 'Incomplete' (0 Orders, 2 Change Requests). On the right, a sidebar lists various categories like 'Requisitions', 'Orders', 'Agreements', 'Deliverables', 'Supply Base', and 'Administration'. Under 'Orders', the 'Manage Orders' option is highlighted with a red box.

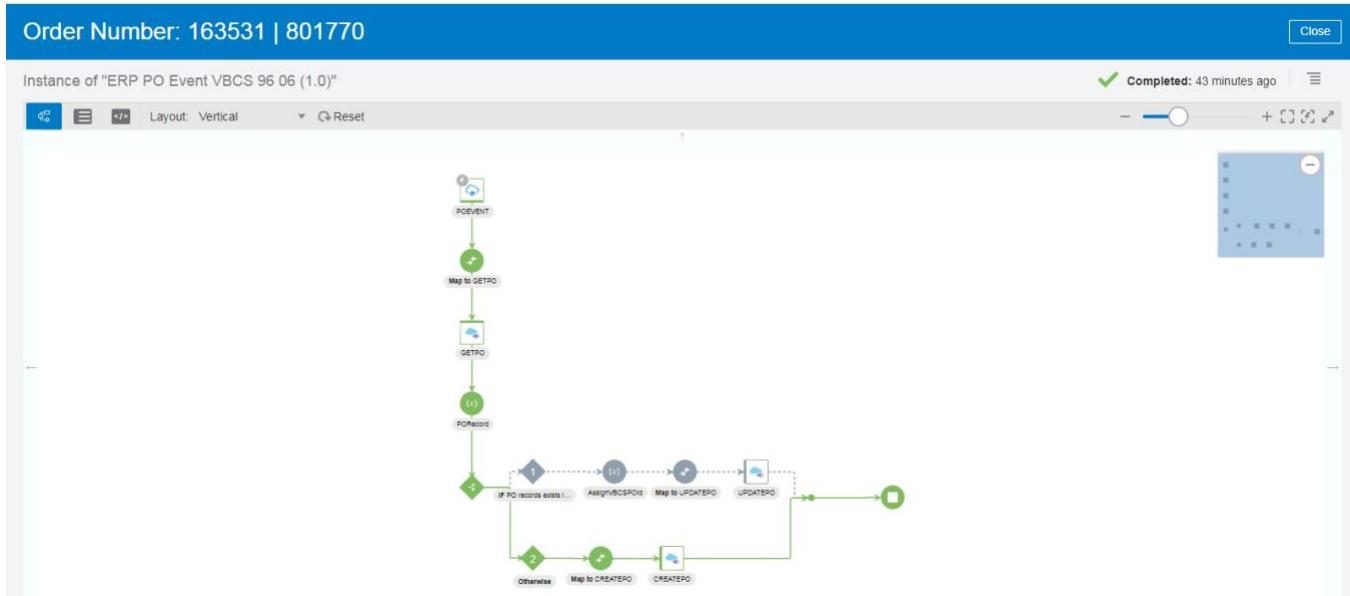
8. Search for the Order Number and wait for the Status field to change to “Open”

The screenshot shows the 'Manage Orders' search interface. The search criteria include 'Order' set to '163531'. The search results table shows one row for Order 163531, which is currently in the 'Open' status. The table includes columns for Order, Description, Supplier, Ordered, Currency, Status, Funds Status, Change Order Funds Status, Life Cycle, Change Order, Creation Date, Supplier Site, Supplier Contact, Acknowledgment Due Date, and Shipping Method.

9. Go back to Oracle Integration -> Monitoring -> Tracking Page and Search for your instance with the Order Number as Tracking identifier

The screenshot shows the Oracle Integration Cloud Monitoring -> Track Instances page. A search bar at the top contains the tracking identifier '163531', which is also highlighted with a red box. Below the search bar, a table provides detailed tracking information for Order 163531, including the Order Number, Instance ID, PO Header ID, Item Description, Received time, Completed time, and Duration.

- Click on the instance and verify that the Flow is completed successfully

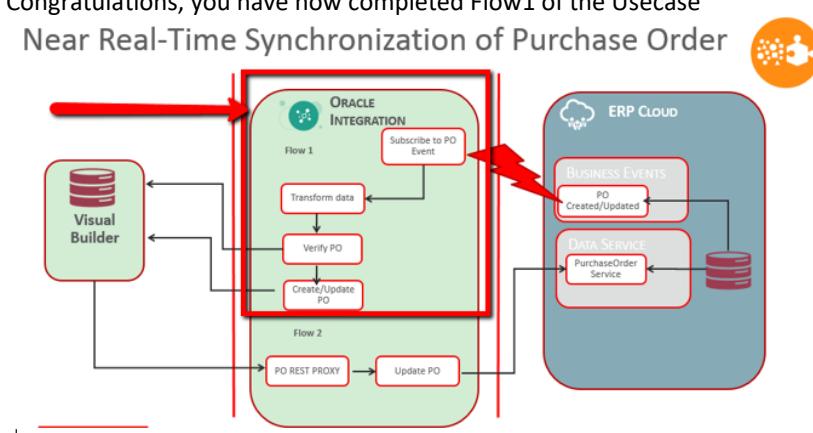


Since it is a new PO record which does not exist in VBCS “Otherwise” flow path is chosen based on the condition

11. Navigate to Visual Builder and Click on imported application
12. Select the Business Objects Tab and click on “PO” object you should see a new record is created

The screenshot shows the Oracle Visual Builder interface with the "Business Objects" tab selected. Under the "PO" object, the "Data" tab is active, displaying a table with one row of data. The columns are: Id (id), LOCIId (LOCIId), OrderNumber (orderNumber), POHeaderId (pOHeaderId), ProcurementBUId (procurementBUId), and ProcurementBusinessUnit (procurementBusinessUnit). The data row is: 63, 163531, 300000176127931, 300000046987012, US1 Business Unit.

Congratulations, you have now completed Flow1 of the Usecase
Near Real-Time Synchronization of Purchase Order



You leveraged rich capabilities of Oracle Integration such as ERP Adapter Event Subscription Capabilities using and App-driven (trigger-based) integration, ERP Cloud adapter, Rest adapter, Data Mapper, Actions such as Invoke and Activities such as Assign, Map, If-Otherwise etc, Configuring Business Tracking Identifiers and monitoring running flows.

You could now leverage this knowledge to design, activate and monitor several use cases for ERP Real time Synchronization

Creating Connections (Flow2)

The following Connections have been created and configured. You will be using these connections for creating Integration flow

Connection Name	Connection Type
PO REST Interface Con 96 06	Rest Adapter
ERP Conn 96 06	ERP Adapter

Creating the PO Proxy Integration (Flow2)

1. On the Oracle Integration home page, click **Integrations**.
2. On the Integrations page, click **Create**. The Create Integration - Select a Style/Pattern dialog is displayed.
3. Select **App Driven Orchestration** type of integration. The **Create New Integration** dialog is displayed.
4. Enter the following information:

Field Element	Value
What do you want to call your integration?	ERP PO Proxy <ClassId> <StudentId> Ex: ERP PO Proxy 96 06
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of 01.00.0000. Or, if you want to change the version number, enter the version using numbers only in this format: xx.xx.xxxx.
What does this integration do?	This integration demonstrates the use of Oracle Integration REST adapter and ERP Cloud Adapter to create a REST to SOAP proxy for ERP Cloud webservices
Which package does this integration belong to?	Leave blank

Create Integration - Select a Style

How would you like to build your integration? Select a style to use.

App Driven Orchestration  Multi-step Integration flow triggered by an Application or API. <input type="button" value="Select"/>	Scheduled Orchestration  Multi-step Integration flow triggered by a Schedule. Commonly used for Batch/Bulk Integrations or File processing. <input type="button" value="Select"/>	File Transfer  Seamlessly and securely move files across the network. <input type="button" value="Select"/>
Basic Routing  Basic App to App Routing with Data Mapping. <input type="button" value="Select"/>	Publish To OIC  Publish messages from Apps to OIC Pub/Sub Channel. <input type="button" value="Select"/>	Subscribe To OIC  Subscribe to messages from OIC Pub/Sub Channel. <input type="button" value="Select"/>

Create New Integration

Create New Integration

Enter information that describes this integration.

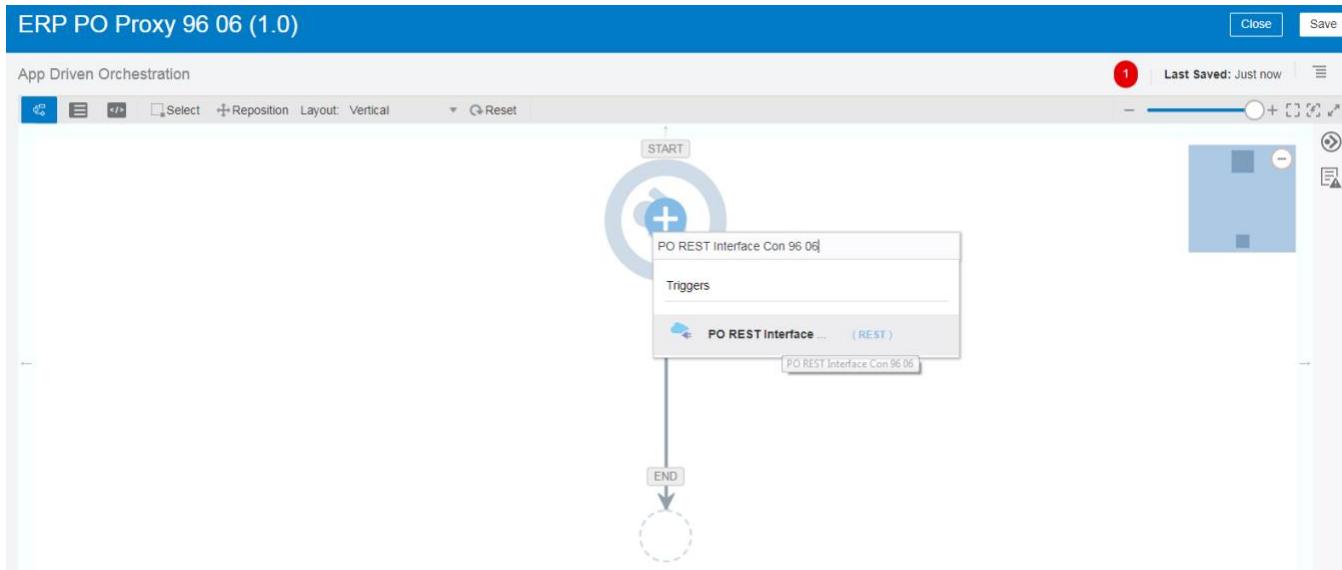
Describe this integration Use a meaningful name and description that will help others find and understand this integration. The Identifier and Version can be set only when the integration is created. The combination of Identifier and Version must be unique.

* What do you want to call your integration?	What does this integration do?
ERP PO Proxy 96 06	This integration is a rest proxy for Purchase order service which will be invoked from <u>VBCS</u> page
* Identifier	Which package does this integration belong to?
ERP_PO_PROXY_96_06	Enter a new or existing package name
* Version	
01.00.0000	

Create **Cancel**

Click **Create**. The integration canvas is displayed

5. Hover over the circle next to Start, and click on the + sign
6. Select the REST Connection (PO REST Interface Con 96 06) from the list which is pre created for you
7. Enter details as in the screenshot below to define the endpoint in the flow



8. In the Basic info page provide the endpoint name and description. Click Next

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the REST adapter

Basic Info

* What do you want to call your endpoint?
POPROXY

What does this endpoint do?
REST interface for PO Service

Select to configure multiple resources or verbs. (maximum 11)

- In the Resource Configuration page provide the operation name, URI, and HTTP verb, and select options for Request/Response payload.

Configure REST Endpoint

Welcome to the REST Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the REST adapter.

Resource Configuration

* Provide an operation name
POPROXY

What does this operation do?
Describe what this operation does

* What is the endpoint's relative resource URI?
/PO_PROXY

* What action do you want to perform on the endpoint?
POST

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

Add and review parameters for this endpoint

Configure a request payload for this endpoint

Configure this endpoint to receive the response

Configure Request Headers? Standard Custom

10. In the Request page Select Request payload format as “JSON Sample” and provide the sample JSON below by clicking on the inline link

```
{
  "id": 24,
  "IOCId": 3,
  "orderNumber": "163521",
  "pOHeaderId": "300000074157561",
  "procurementBUId": "300000046987012",
  "procurementBusinessUnit": "US1 Business Unit",
  "soldToLegalEntity": "US1 Legal Entity",
  "supplier": "Dell.",
  "supplierId": "300000047414679"
}
```

11. In the Response page Select Response payload format as “JSON Sample” and provide the sample JSON below by clicking on the inline link

```
{
  "pOHeaderId": "300000074157561",
  "orderNumber": "162180",
  "soldToLegalEntityId": "300000074157561",
  "changeOrderNumber": 3,
  "requestStatus": "SUCCESS"
}
```

12. Review the Summary page and click on Done

Configure REST Endpoint

REST Endpoint Configuration Summary
REST endpoint configuration was successful.

Basic Info

Request Parameters

Request

Request Headers

CORS Configuration

Response

Response Headers

Summary

Rest endpoint summary

POPROXY

Description

Endpoint Summary

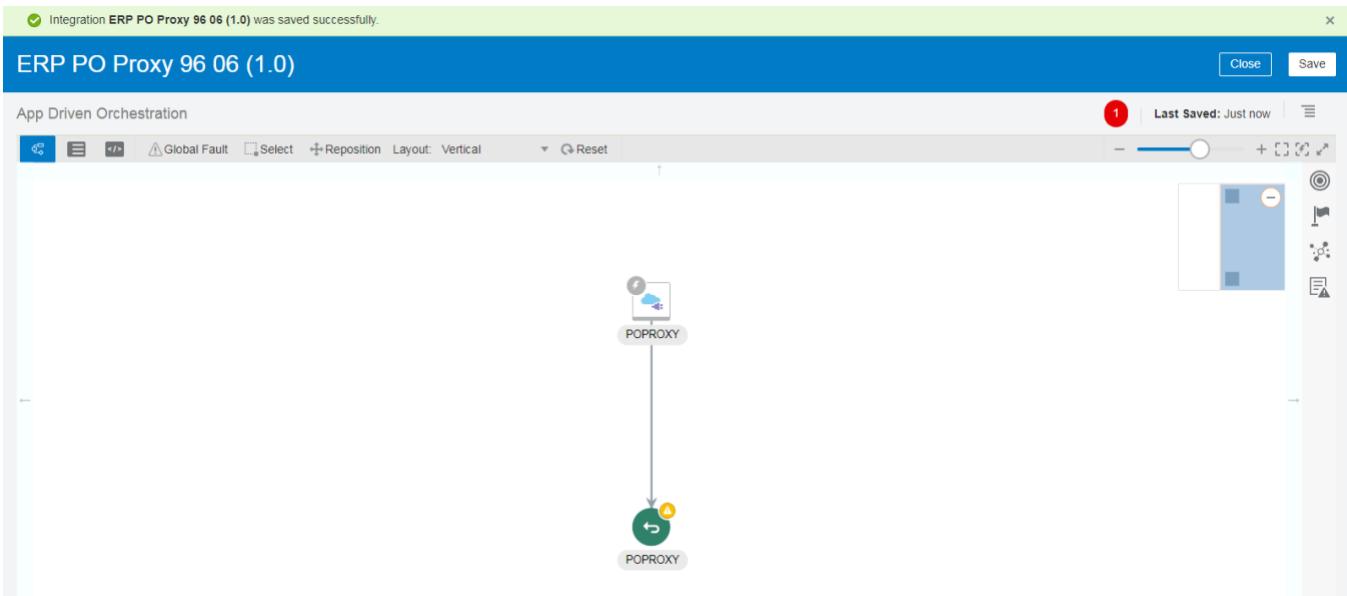
REST Service URI: /PO_PROXY

Method: POST

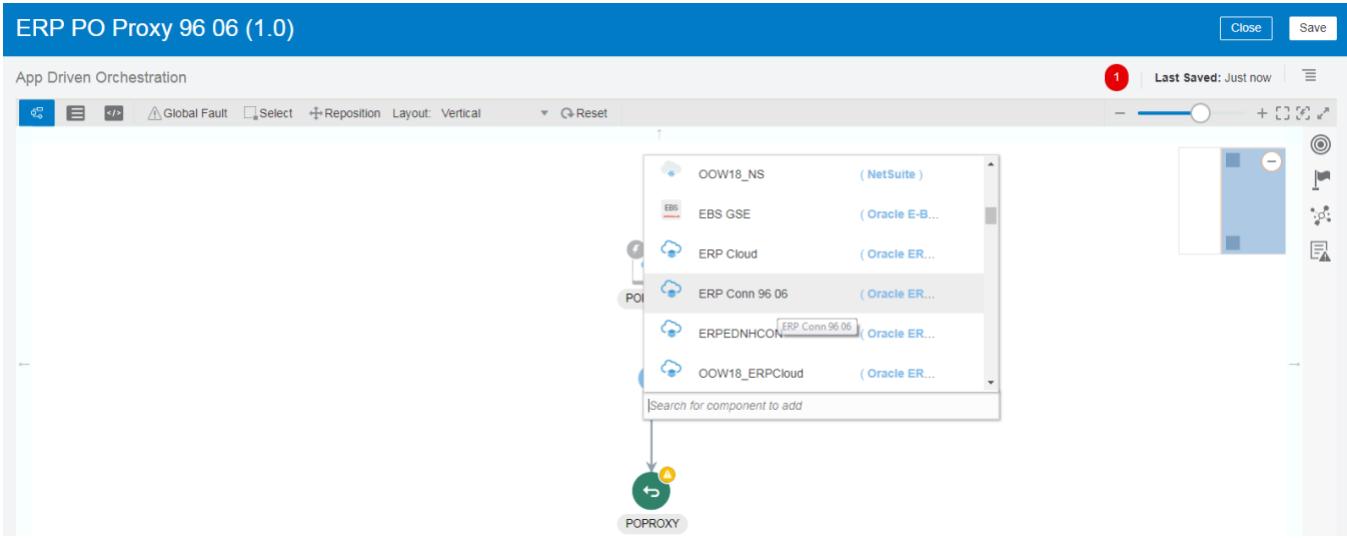
Request Media Type: JSON

Response Media Type: JSON

A Request/Response Flow is created. Delete the “Map to POPROXY” Action



13. Hover on the wiring next to PROXY Action and Click on "+" sign and add the "ERP Conn 96 06" connection which is pre created



14. In the Basic info page provide endpoint name , endpoint description and Click Next

Configure Oracle ERP Cloud Endpoint

Welcome to the Oracle ERP Cloud Endpoint Configuration Wizard
This wizard helps you configure an endpoint using the Oracle ERP Cloud connection. You will be asked to specify configuration parameters and define an operation for the service.

Basic Info

- Actions
- Operations
- Response
- Summary

* What do you want to call your endpoint?
changePO

What does this endpoint do?
endpoint to update Purchase order

Describe the endpoint's purpose and detail

15. In the Actions Page, Select the Radio Button “Query, Create, Update or Delete Information” to consume Purchase Order V2 Service

Configure Oracle ERP Cloud Endpoint

Welcome to the Oracle ERP Cloud Endpoint Configuration Wizard
This wizard helps you select one of the capabilities of Oracle ERP Cloud Adapter

Basic Info

- Actions
- Operations
- Response
- Summary

What would you like to do with Oracle ERP Cloud Adapter?

Query, Create, Update or Delete Information
Perform operations such as Find Catalog, Create Orders, Update Accounts, Process Expenses etc

Import Bulk Data into Oracle ERP Cloud
Perform FBDI-compliant bulk operations such as Import Asset Leases, Journal Entries, Bank Statements, Payable Invoices, Project Tasks, Sales Orders, Shipping Transactions etc. You can also configure event notifications and callbacks on these operations.

Send Files to ERP Cloud
Upload files to Universal Content Management (UCM) in ERP Cloud. Note that once the files are uploaded, you will need to add a separate action in your integration flow to invoke the appropriate scheduled process or API in ERP Cloud to process the file. Use this option only when you cannot use the Import Bulk Data option above which automatically takes care of all this.

[Learn more about these Actions](#)

16. In the Operations Page Select Browse By -> Services and Search for Purchase Order.
Select PurchaseOrderService and changePurchaseOrder operation to consume in our flow.
Note: Select the correct PurchaseOrderService which has “acknowledgePurchaseOrder” in the list of operations

Configure Oracle ERP Cloud Endpoint

 Help ▾ < Back Next > Cancel Done



Configure the Operations to Perform in the Target Oracle ERP Application
Select the business object or service and operation to use for the target integration.

X

- Basic Info
- Actions
- Operations

Response

Summary

Browse by Services ▾

Select a Service

Purchase

All ▾

- PurchaseAgreementService
- PurchaseOrderService
- PurchaseOrderService
- PurchaseRequestService

Select the Operation to Perform on the Business Object/Resource or Service

changePurchaseOrder ▾

Lifecycle: Active

Description

Click Next

17. Review the Summary Page and Click Done

Configure Oracle ERP Cloud Endpoint

 Help ▾ < Back Next > Cancel Done



Oracle ERP Cloud Endpoint Configuration Summary
Oracle ERP Cloud endpoint configuration was successful.

X

- Basic Info
- Actions
- Operations

Response

Summary



changePO

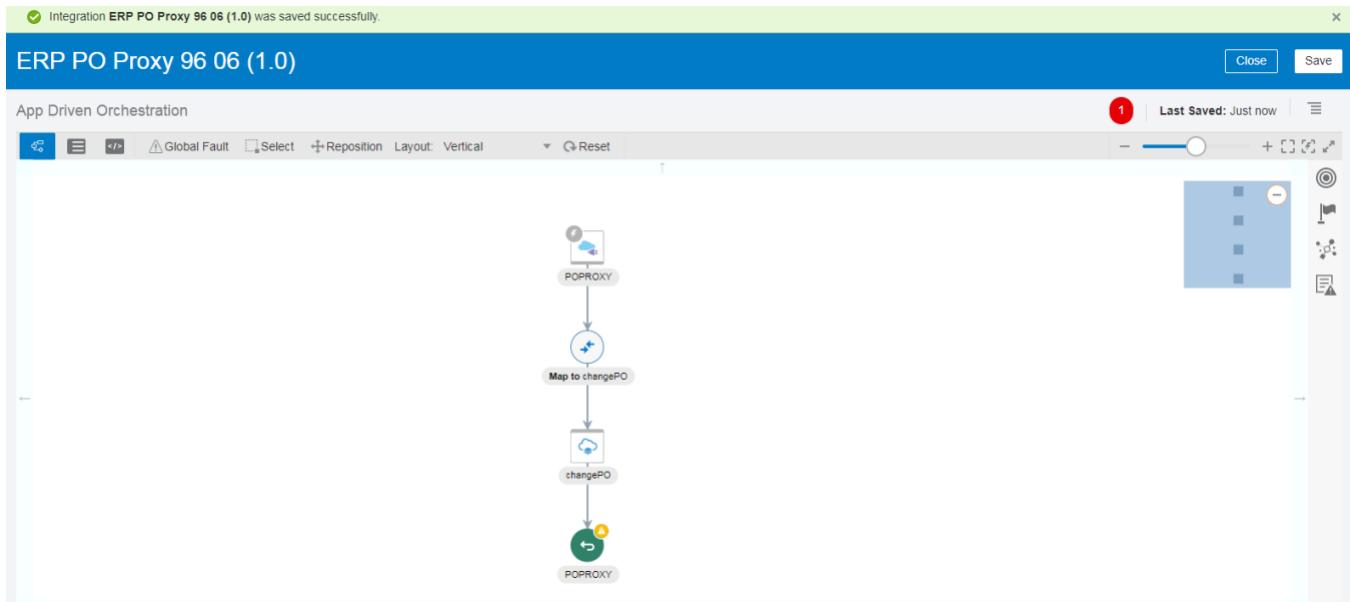
Description

endpoint to update Purchase order

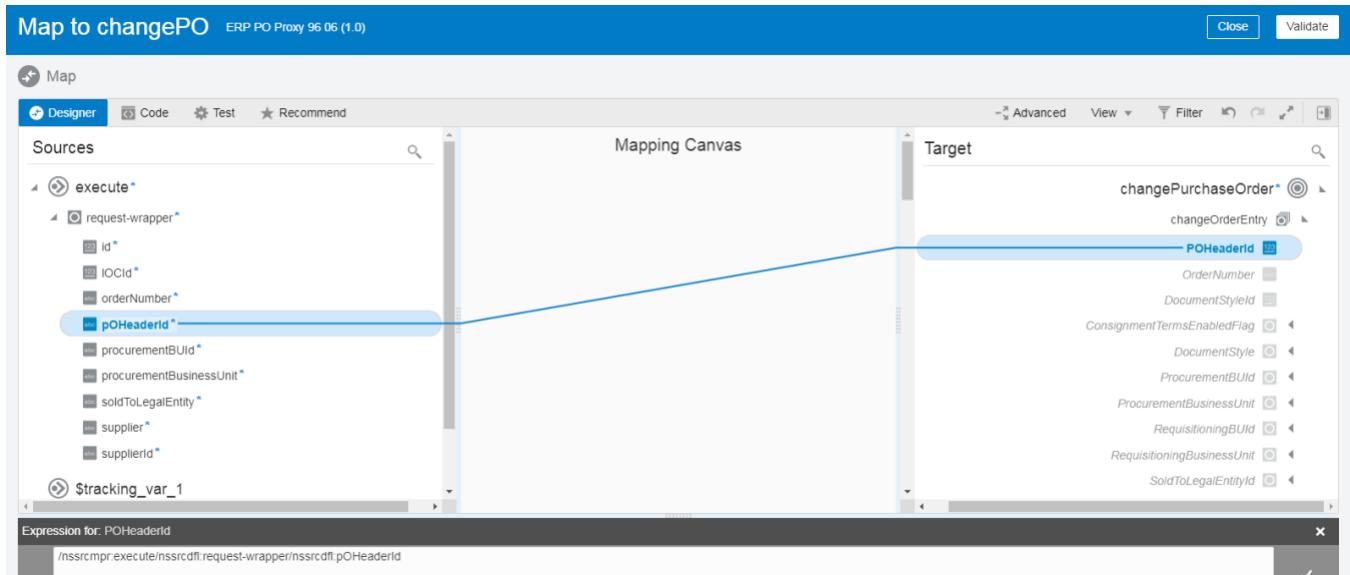
Business Object PurchaseOrderService

Operation changePurchaseOrder

18. An invoke Action is configured with changePO operation and an empty Map (Map to changePO) Action is created



19. Edit the Map (Map to changePO) Action and provide below mapping



Right-click the element `changeOrderEntry -> ChangeOrderDescription` and select “Create Target Node” which opens up an expression window. Enter ‘Update of LOClientId’ (make sure to include the single-quotes). Click the check mark button.

Map to changePO ERP PO Proxy 96 06 (1.0)

Map

Designer Code Test Recommend Advanced View Filter

Sources

- execute*
 - request-wrapper*
 - id*
 - IOCId*
 - orderNumber*
 - pOHeaderId*
 - procurementBUID*
 - procurementBusinessUnit*
 - soldToLegalEntity*
 - supplier*
 - supplierId*
- \$tracking_var_1

Mapping Canvas

Target Fields:

- NoteToReceiver
- NoteToSupplier
- TaxationCountryCode
- TaxationCountry
- DocumentFiscalClassificationCode
- DocumentFiscalClassification
- InterfaceSourceCode
- ReferenceNumber
- ChangeOrderDescription **A**
- ChangeOrderInitiatingParty
- ChangeOrderInitiatorId
- ChangeOrderInitiatorEmail
- ChangeOrderInitiatorName
- FirstPartyTaxRegistrationId
- FirstPartyTaxRegistrationNumber

Expression for: ChangeOrderDescription

```
"Update of LOC Id"
```

Expand “HeaderFlexField” element and map the following fields from Source to Target

Map \$Id -> \$customerPO

Map to changePO ERP PO Proxy 96 06 (1.0)

Map

Designer Code Test Recommend Advanced View Filter

Sources

- execute*
 - request-wrapper*
 - id*
 - IOCId*
 - orderNumber*
 - pOHeaderId*
 - procurementBUID*
 - procurementBusinessUnit*
 - soldToLegalEntity*
 - supplier*
 - supplierId*
- \$tracking_var_1

Mapping Canvas

Target Fields:

- FirstPartyTaxRegistrationId
- FirstPartyTaxRegistrationNumber
- ThirdPartyTaxRegistrationId
- ThirdPartyTaxRegistrationNumber
- PurchaseOrderEntryLine
- HeaderFlexfield
- PoHeaderId
- customerPo **B**
- @nil
- warrantyExpirationDate
- locId
- ackFlag
- _FLEX_Context
- _FLEX_Context_DisplayValue

Expression for: customerPo

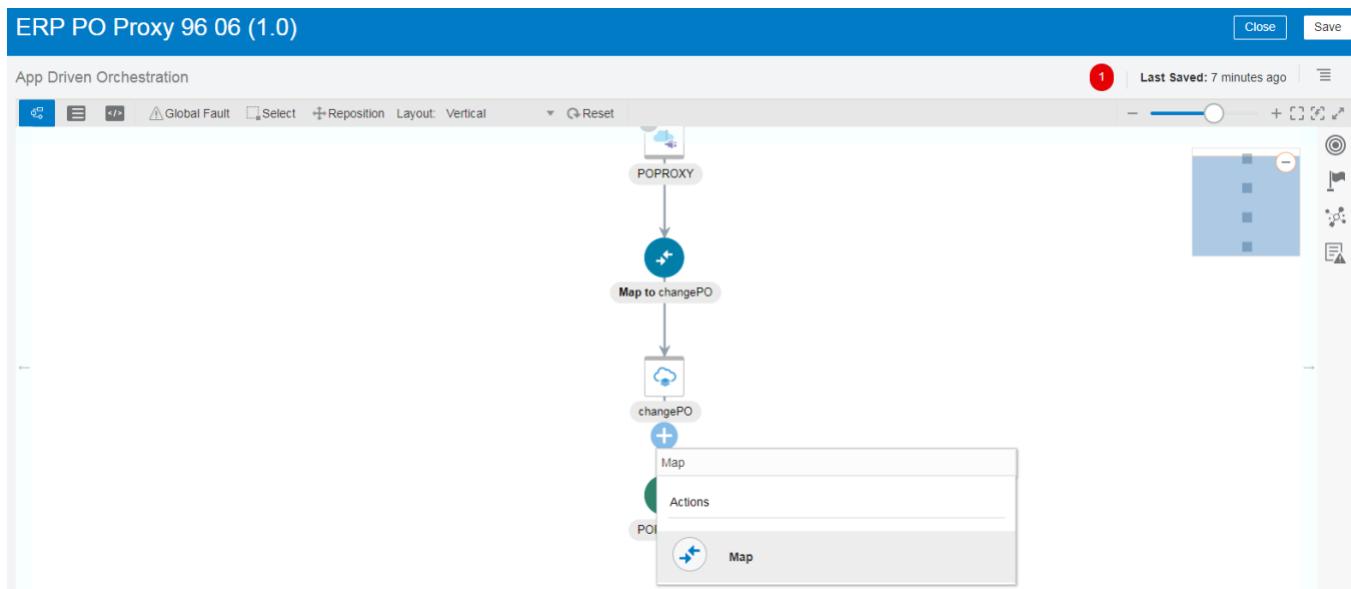
```
/hssrcmpr.execute/hssrcdff.request-wrapper/hssrcdff.id
```

Map IOClId -> locId

The screenshot shows the Oracle Integration Map tool interface. On the left, under 'Sources', there is a tree view for the 'execute*' action, with several fields listed: id*, locId*, orderNumber*, pOHeaderId*, procurementBUID*, procurementBusinessUnit*, soldToLegalEntity*, supplier*, and supplierId*. On the right, under 'Targets', there is a tree view for the 'changePO' action, with fields: ThirdPartyTaxRegistrationId, ThirdPartyTaxRegistrationNumber, PurchaseOrderEntryLine, HeaderFlexfield, PoHeaderId, customerPo, warrantyExpirationDate, locId, @nil, ackFlag, _FLEX_Context, _FLEX_Context_DisplayValue, and _FLEX_NumOfSegments. A blue line connects the 'locId*' field in the source to the 'locId' field in the target. Below the mapping canvas, an expression bar shows the path: /nssrcmpr:execute/nssrcdf:request-wrapper/nssrcdf:locId.

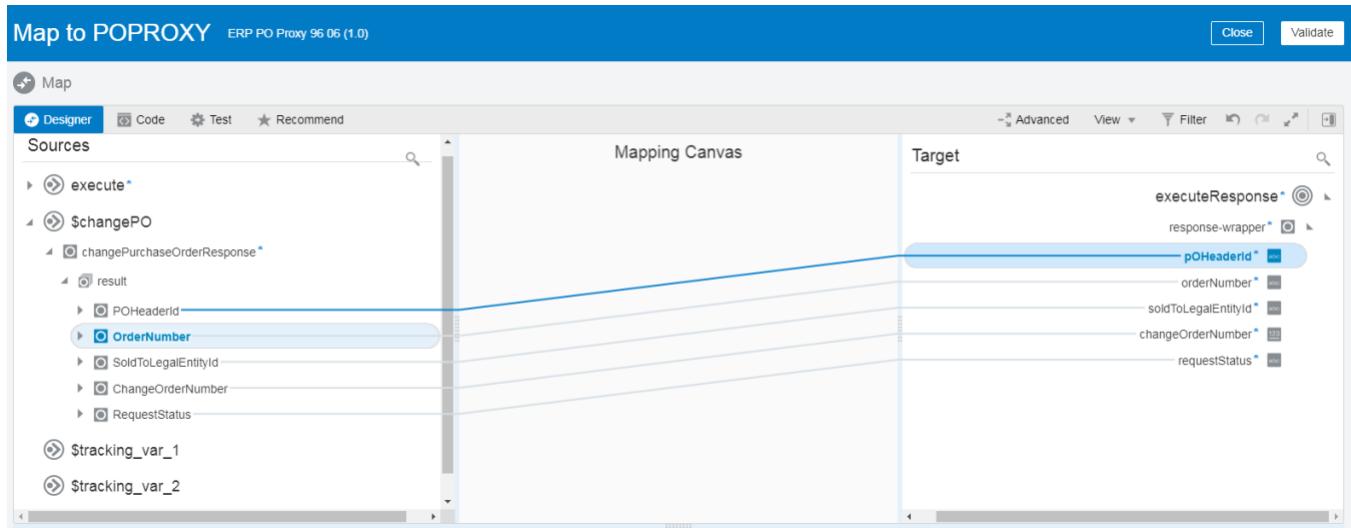
Validate and Save your integration

20. Hover on the wiring next to the changePO invoke Action and Click + sign to add a Map Action



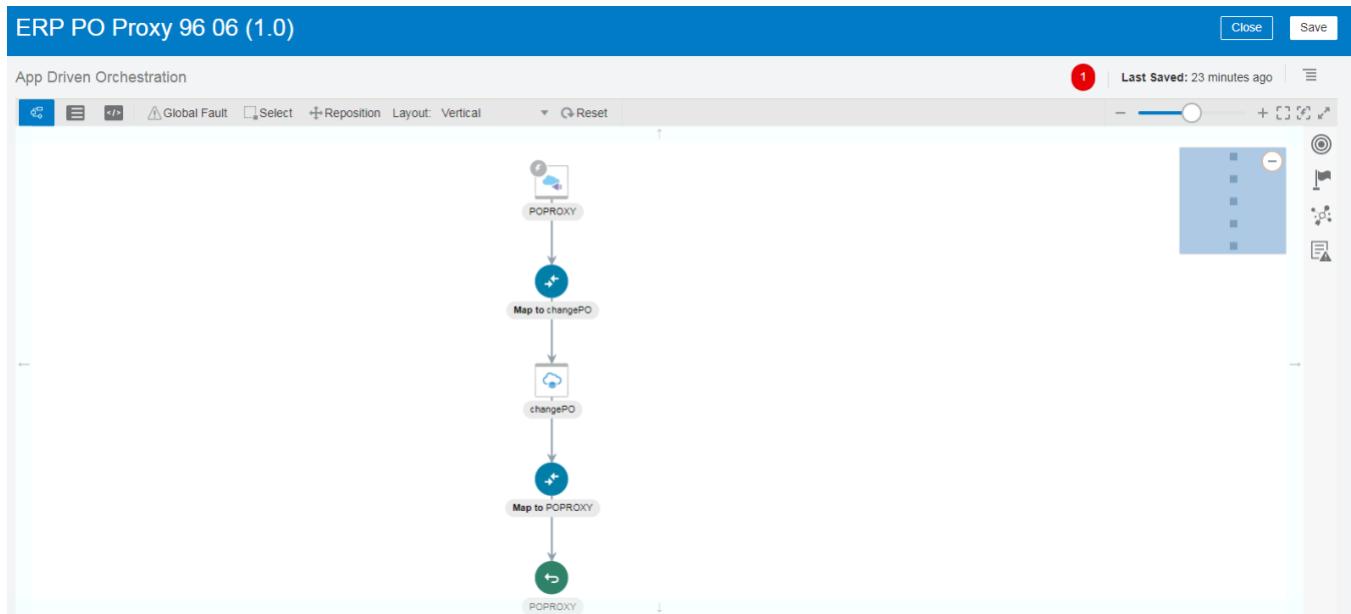
21. Edit the “Map to POProxy” Action and perform the mapping as per below

changePurchaseOrderResponse -> result->POHeaderId **TO** response-wrapper -> poHeaderId
 changePurchaseOrderResponse -> result->OrderNumber **TO** response-wrapper -> orderNumber
 changePurchaseOrderResponse -> result->SoldtoLegalEntityId **TO** response-wrapper -> soldToLegalEntityId
 changePurchaseOrderResponse -> result->ChangeOrderNumber **TO** response-wrapper -> changeOrderNumber
 changePurchaseOrderResponse -> result->RequestStatus **TO** response-wrapper -> requestStatus



Click on Validate and Close

22. Your Final integration flow...



23. Click on Top right Corner and select Tracking. Add the below Business Identifiers for Tracking your instance

Business Identifiers For Tracking

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text
<input checked="" type="checkbox"/>	orderNumber	order Number	tracking_var_1	How to track it?
<input checked="" type="checkbox"/>	id	id	tracking_var_2	How to track it?
<input checked="" type="checkbox"/>	IOClId	IOC Id	tracking_var_3	How to track it?

Save Cancel

Click on Save.

Activate Integration (Flow2)

- On the Integrations page, click on the “ERP PO Proxy <ClassId> <StudentId>” flow using the slider button and Activate the Integration

Activate Integration

ERP PO Proxy 96 06 (1.0)

Publish: This integration can be published to API Platform CS. To publish it now, click "Activate and Publish...". You can also publish it later.

Oracle Recommends

Contribute integration mappings to Oracle Recommends.

Learn More

Oracle Asserter: When Asserter recording is enabled, payloads will be captured and integration instances will be recorded. Recordings can be played later and maximum five recordings will be maintained for an integration.

Enable Asserter Recording

Tracing: When tracing is enabled, integration activity can be viewed in the Activity Stream.

Enable tracing

Include payload

Warning: When payload is included, sensitive information from the payload is written into log files, which can be downloaded and viewed. This may pose a security risk, and also slow down your system. Not recommended in a production environment.

Learn More

Activate and Publish... Activate Cancel

2. Click on How to run next to the Activation bar

The screenshot shows the Oracle Integration Cloud Designer interface. On the left, there's a sidebar with options like Designer, Integrations, Connections, Lookups, Packages, Agents, Adapters, and Libraries. The main area is titled 'Integrations' and shows a list with one item: 'ERP PO Proxy 96 06 (1.0)'. Below it, it says 'APP DRIVEN ORCHESTRATION' and 'This integration is a rest proxy for Purchase order service which will be invoked from VBCS page'. A tooltip is overlaid on the 'How to run' button, containing the endpoint URL and other navigation links.

Open the metadata url generated in a browser window.

This metadata url provides all the description about your REST Endpoint

Endpoint Description

Endpoint URL

https://OICPMDemoUMic-oicpm.uscom-central-1.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY_96_06/1.0/PO_PROXY

Swagger

https://OICPMDemoUMic-oicpm.uscom-central-1.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY_96_06/1.0/metadata/swagger

How to Run

http://www.oracle.com/pls/topic/lookup?ctx=oic_en&id=ICSUG-GUID-205B916C-1075-4603-A9E2-72A6C8C4AB3C

Resource /PO_PROXY

Method POST

Request

Request Media Type

- application/json

Request sample

```
{ "id": 24, "IOCID": 3, "orderNumber": "163521", "pOHeaderId": "300000074157561", "procurementBUId": "300000046987012", "procurementBusinessUnit": "US1 Business Unit", "soldToLegalEntity": "US1 Legal Entity", "supplier": "Dell", "supplierId": "300000047414679" }
```

Response

Response Media Type

- application/json

Response sample

```
{ "pOHeaderId": "300000074157561", "orderNumber": "162180", "soldToLegalEntityId": "300000074157561", "changeOrderNumber": 3, "requestStatus": "SUCCESS" }
```

3. Copy the Swagger url from the Endpoint description. We will use the swagger definition in next section to create a service connection in VBCS Web application

Invoke ERP Cloud Service from VBCS Web App

1. Select the Visual Builder tab, from the list of applications open the **LetterOfCreditPortal<ClassId><StudentId> Web application**
2. On the left hand side select Service Connections tab

The screenshot shows the Oracle Visual Builder interface. On the left, there's a sidebar with a red box around the 'Service Connection' icon. The main area has tabs for 'Iclistpage' and 'editpage'. A modal window titled 'ERROR' displays the message 'Could not load data: status 404'. To the right, there's a form with fields like 'Id', 'LOC Id', 'POHeaderId', etc., all currently empty.

3. Click on the “+ Service Connection” to create a new Service Connection to invoke on a button click
4. In the “Create Service Connection” page select “Define by Specification”

The screenshot shows the 'Create Service Connection' dialog. It has three options: 'Select from Catalog' (green), 'Define by Specification' (orange, highlighted with a red box), and 'Define by Endpoint' (purple). Each option has a brief description below it. A 'Cancel' button is visible at the bottom right.

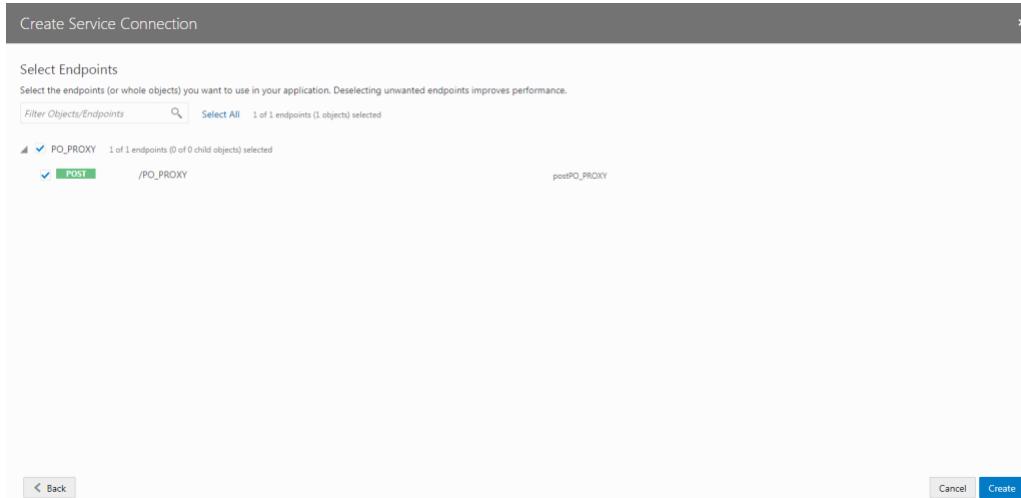
5. Provide the highlighted information as per the screenshot and Click Next

The screenshot shows the 'Create Service Connection' dialog for 'Define by Specification'. Several fields are highlighted with red boxes and arrows pointing to callouts:

- API Type:** A dropdown menu with 'Swagger' selected, indicated by a red arrow.
- Service Specification:** A radio button group where 'Web Address' is selected, indicated by a red arrow.
- Service Id:** A text input field containing the URL: https://OICPMDEMOUic-oicpm.uscom-central-1.oraclecloud.com:443/ic/api/integration/v1/flows/rest/ERP_PO_PROXY, indicated by a red arrow with the note: "Provide the swagger url copied in the previous steps".
- Authentication Mechanism:** A dropdown menu with 'Basic' selected, indicated by a red arrow with the note: "Provide your Oracle Integration Credentials".
- Username:** A text input field containing 'kishore.s.katta@oracle.com', indicated by a red box.
- Password:** A password input field containing '*****', indicated by a red box.

At the bottom right, there are 'Cancel' and 'Next >' buttons.

6. In the Select Endpoints page Select the “PO_PROXY” endpoint and click on Create

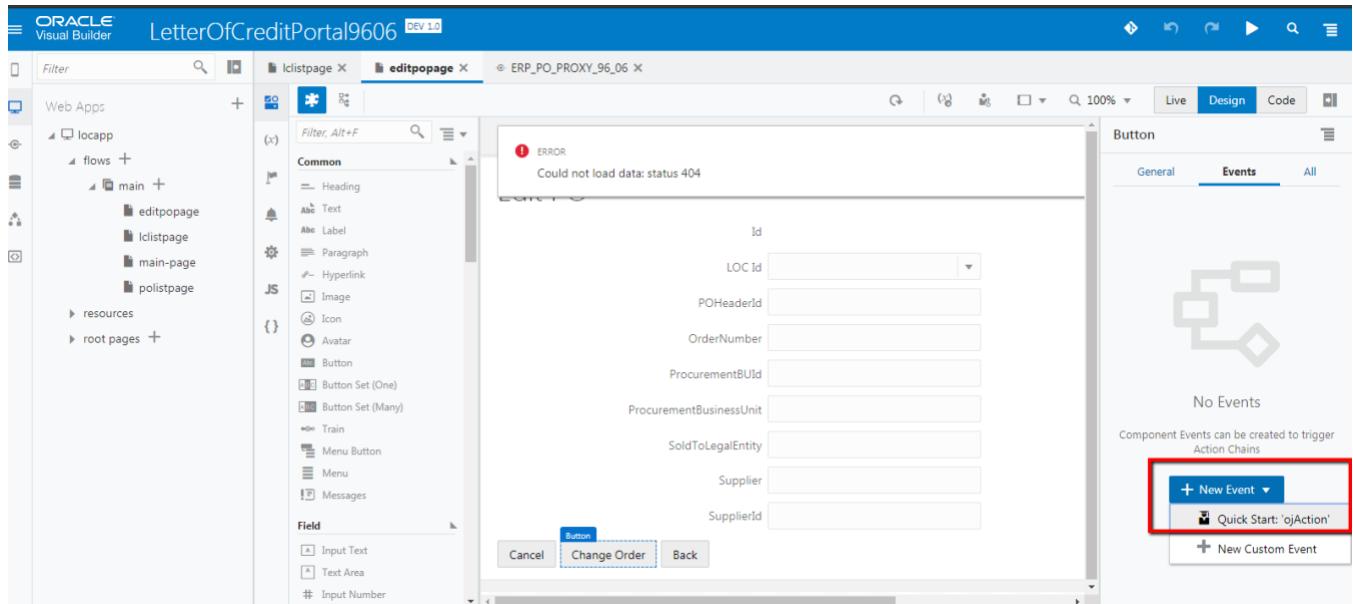


Connection to the endpoint is created

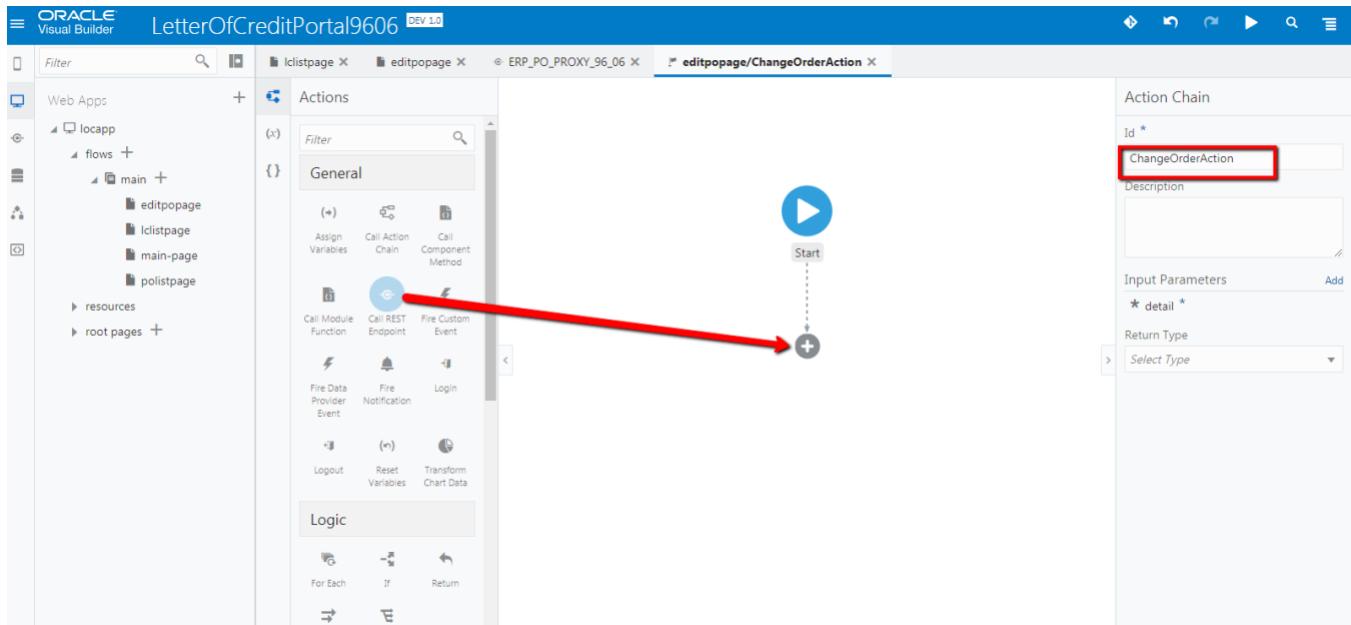
7. (1) Click on Web application tab and (2) select the “editpopage” by expanding the localapp->flows->main. (3) Select the “Change Order” button on the web page. (4) On the right handside select the Events tab

Oracle Product Management

8. Click on New Event and select “Quick Start: ojAction” to create a new Action Chain



9. Provide Id as “ChangeOrderAction”. Drag and drop “Call REST Endpoint” action from Actions palette onto to the “+ Sign”



The screenshot shows the Oracle Visual Builder interface. On the left, there's a sidebar with a tree view of 'Web Apps' containing 'locapp' with 'flows' like 'editpage', 'listpage', 'main-page', and 'polistpage'. Below that are 'resources' and 'root pages'. In the center, a large panel titled 'Actions' contains various icons for actions like 'Assign Variables', 'Call Action Chain', 'Call Component Method', 'Call Module Function', 'Call REST Endpoint', 'Fire Custom Event', 'Fire Data Provider Event', 'Fire Notification', 'Logout', 'Reset Variables', and 'Transform Chart Data'. Below these are sections for 'Logic' with 'For Each', 'If', and 'Return'. On the right, a canvas shows a process flow starting with a 'Start' node, followed by a 'Call REST Endpoint' node (which is highlighted with a red box), and ending with a plus sign node. To the right of the canvas, a message says 'No endpoint selected' with a 'Select Endpoint' button.

10. Click on "Select Endpoint" and Configure the Service Connection created earlier.

The screenshot shows a modal dialog titled 'Select Endpoint'. At the top is a 'Filter' input field and a search icon. Below it is a tree view of service connections: 'Business Objects', 'Service Connections', 'ERP_PO_PROXY_96_06', 'PO_PROXY', and 'Process Objects'. Under 'ERP_PO_PROXY_96_06', the 'POST /PO_PROXY' endpoint is selected and highlighted with a green bar. At the bottom right are 'Cancel' and 'Select' buttons.

Click on Select. The endpoint is now configured to be invoked from the action chain.

11. Select “Parameters” -> body which opens a mapper interface page

Call REST Endpoint: callRestEndpoint1

Sources Target

Expression for: body

Cancel Save

12. We want to Map the “editpopage” page variables to the endpoint interface.

Map \$poRecord(Sources) TO body (Target)

Call REST Endpoint: callRestEndpoint1

Sources Target

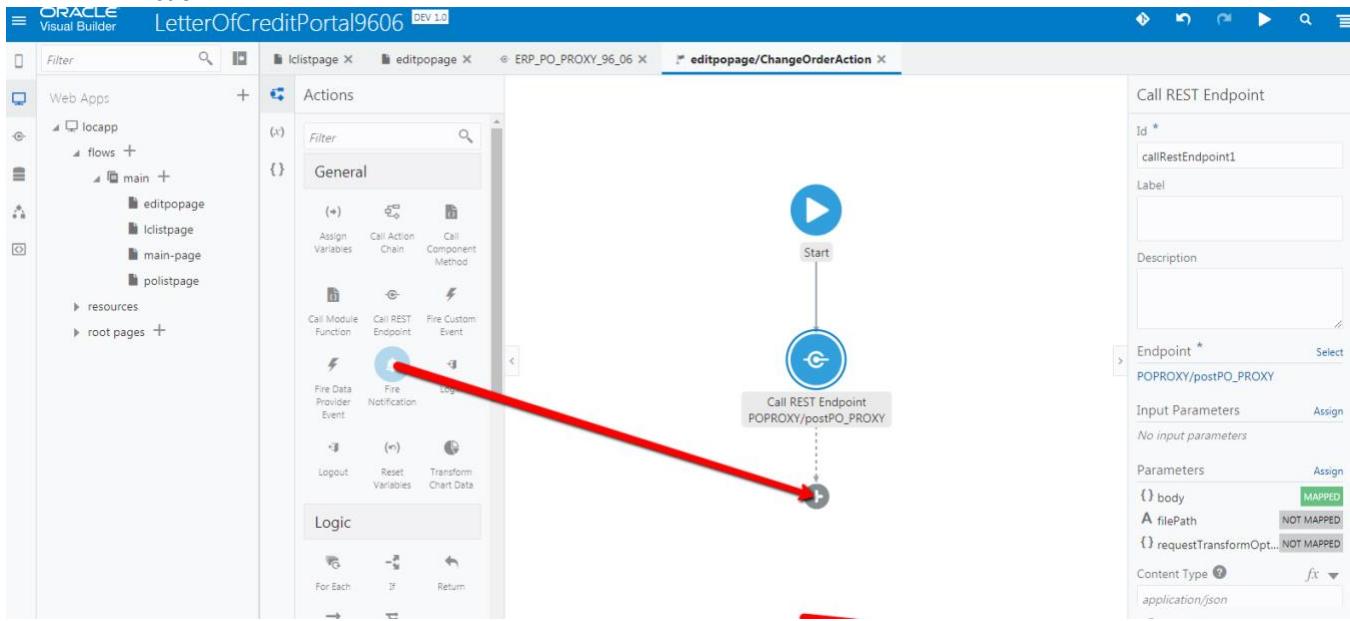
Expression for: body

Cancel Save

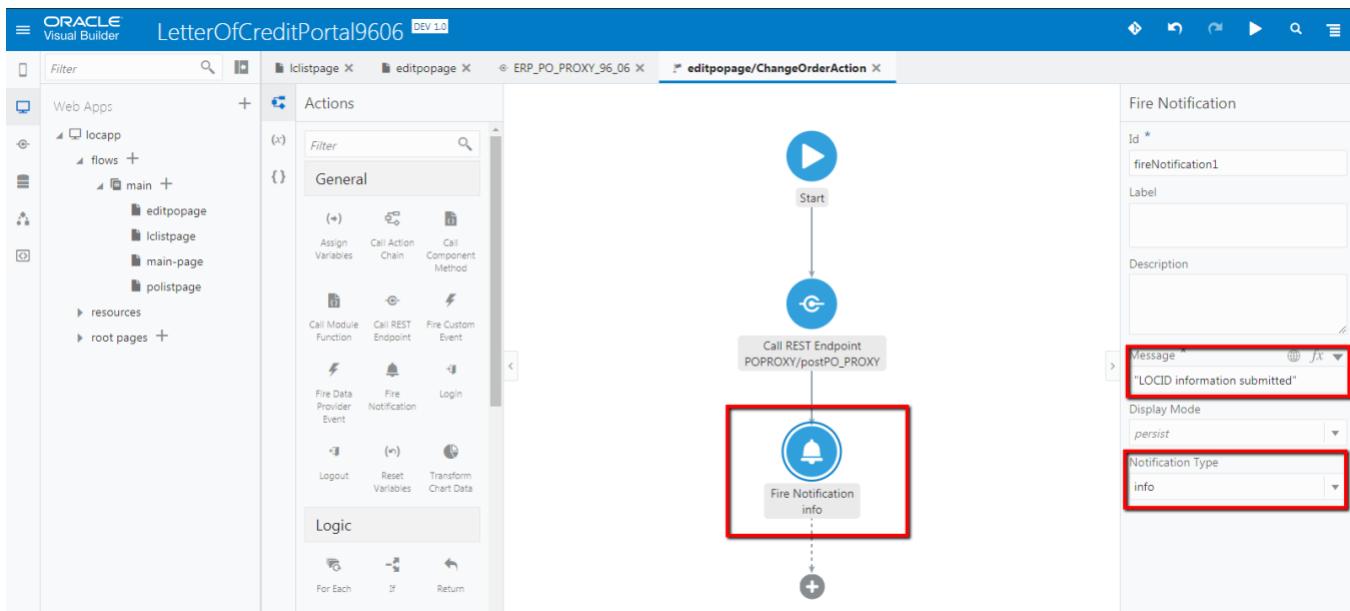
An expression is generated in the expression window based on the mapping. poRecord is a page variable which holds the data from the “editpopage”. Click on Save. You should see the body parameter now marked as “**MAPPED**”

Oracle Product Management

13. From the Action palette Select “Fire Notification” action on to the “+” sign next to the “Call Rest Endpoint” Action

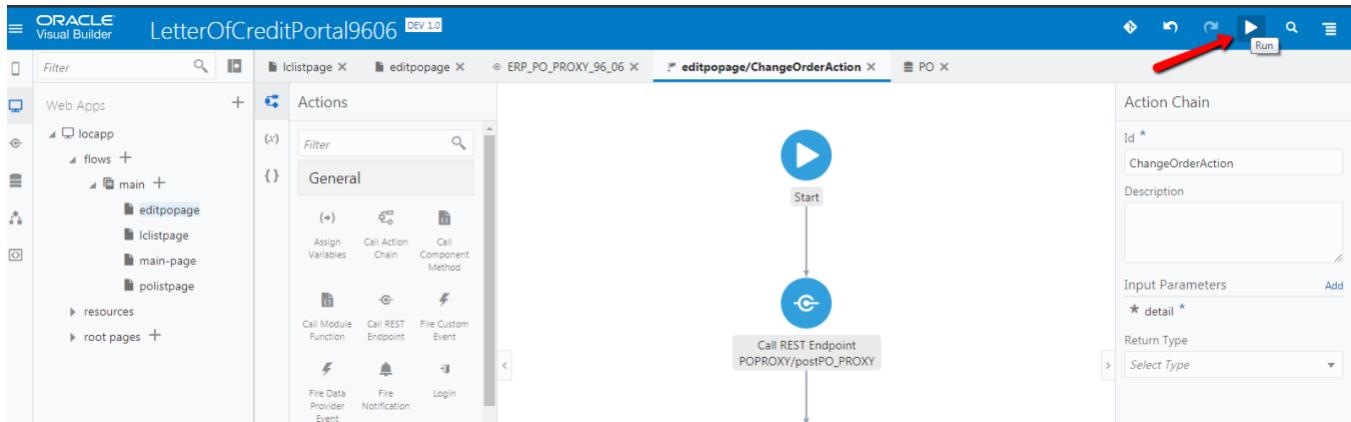


14. Change the Notification type -> info and provide a Message in the text area



Running the VBCS Web App

1. Click on Run



2. The Web application opens up in a separate browser window. We have given a basic web app for you to test the end to end flow

The screenshot shows a simple web application window titled 'LC Portal'. At the top, it has a header with the email 'kishore.x.katta@oracle.com'. Below the header are two buttons: 'LC List Page' and 'PO List Page'.

3. Click on LC List Page which displays all the Letter of Credits Available in a Custom table. Go back to the main page

The screenshot shows a table titled 'LC Portal' displaying 'Letter of Credit' records. The columns are: LOClId, LOCAmount, LOCBank, LOCExpiry, LOCOpeningDate, LOCStatus, and LOCType. The data is as follows:

LOClId	LOCAmount	LOCBank	LOCExpiry	LOCOpeningDate	LOCStatus	LOCType
1	10000	ABC Bank	2020-01-01	2019-01-01	Active	Type1
2	20000	XYZ Bank	2020-01-01	2019-01-01	Active	Type2
3	25000	LVB Bank	2020-01-01	2019-01-01	Active	Type1
4	30000	OCOCO Bank	2020-01-01	2019-01-01	Active	Type2
5	40000	Green Bank	2020-01-01	2019-01-01	Active	Type3
6	50000	Others Bank	2020-01-01	2019-01-01	Active	Type4

At the bottom left, there is a 'Back' button.

4. Click on PO List Page . All the PO records that was created earlier in ERP Cloud was synced into VBCS

The screenshot shows a table titled 'LC Portal' displaying 'Purchase Order' records. The columns are: POHeaderId, OrderNumber, Supplier, and LOClId. The data is as follows:

POHeaderId	OrderNumber	Supplier	LOClId
300000176127931	163531	Dell Inc.	

At the top left, there are buttons for 'Edit PO' and 'Back'. At the bottom left, there is a 'Back' button.

5. Select PO that you created and Click on “Edit PO”

LC Portal

Edit PO	Back
POHeaderId	300000176127931
OrderNumber	163531
Supplier	Dell Inc.
LOCId	

6. In the “Edit PO” page provide “LOCId” information. Leave the rest of the parameters intact

LC Portal kishore.x.katta@oracle.com

Edit PO

Id	63
LOC Id	4
POHeaderId	300000176127931
OrderNumber	163531
ProcurementBUId	300000046987012
ProcurementBusinessUnit	US1 Business Unit
SoldToLegalEntity	US1 Legal Entity
Supplier	Dell Inc.
SupplierId	300000047414679

[Cancel](#) [Change Order](#) [Back](#)

7. Click on “Change Order” button. Which invokes your “ERP PO Proxy <ClassId> <StudentId>” Integration flow

8. You should see an instance is created. Go to Monitoring -> Integrations -> Tracking

Monitoring [Home](#) ORACLE Integration Cloud

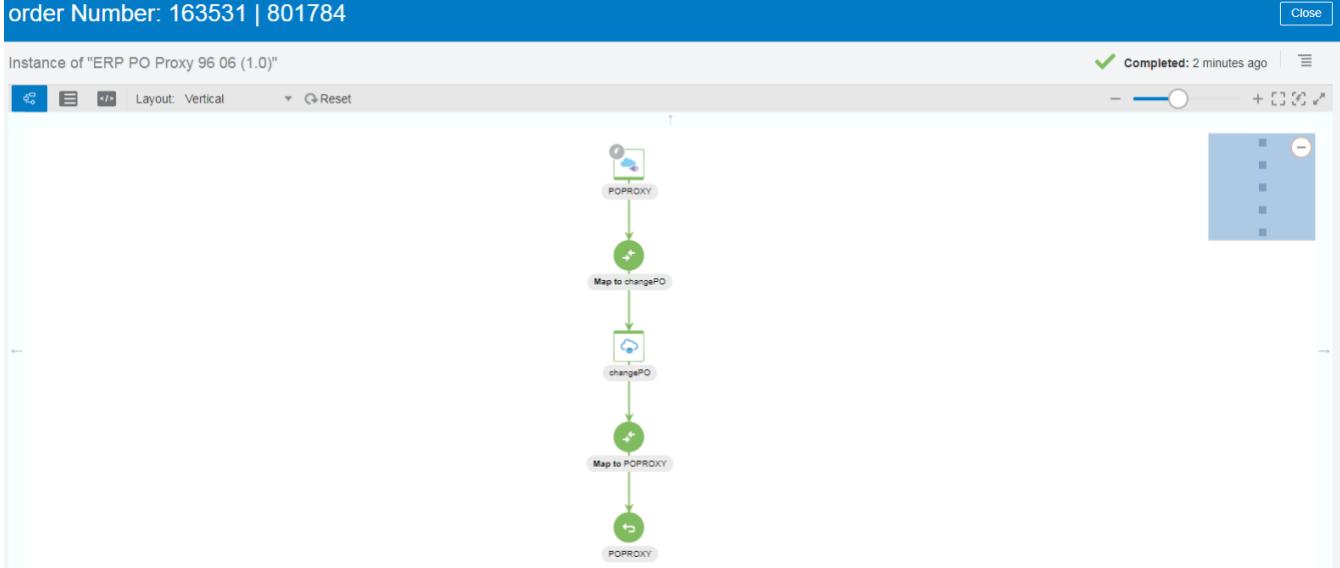
Track Instances

Jan 26, 2019 9:59:09 AM UTC Last 1 Hour

order Number:	Instance ID:	Started	Completed
163531	801784	just now	just now
ERP PO Proxy 96 06 1.0			
COMPLETED			
	id 63	LOC Id 4	Duration 33 sec

9. Select the Order Number and it confirm that the flow is completed and green

order Number: 163531 | 801784



10. Also, observe a notification created in the “Edit PO Page”. Confirming the payload is submitted to the Proxy Integration flow

INFORMATION
"LOCID information submitted"

Id	63
LOC Id	4
POHeaderId	300000176127931
OrderNumber	163531
ProcurementBUID	300000046987012
ProcurementBusinessUnit	US1 Business Unit
SoldToLegalEntity	US1 Legal Entity
Supplier	Dell Inc.
SupplierId	300000047414679

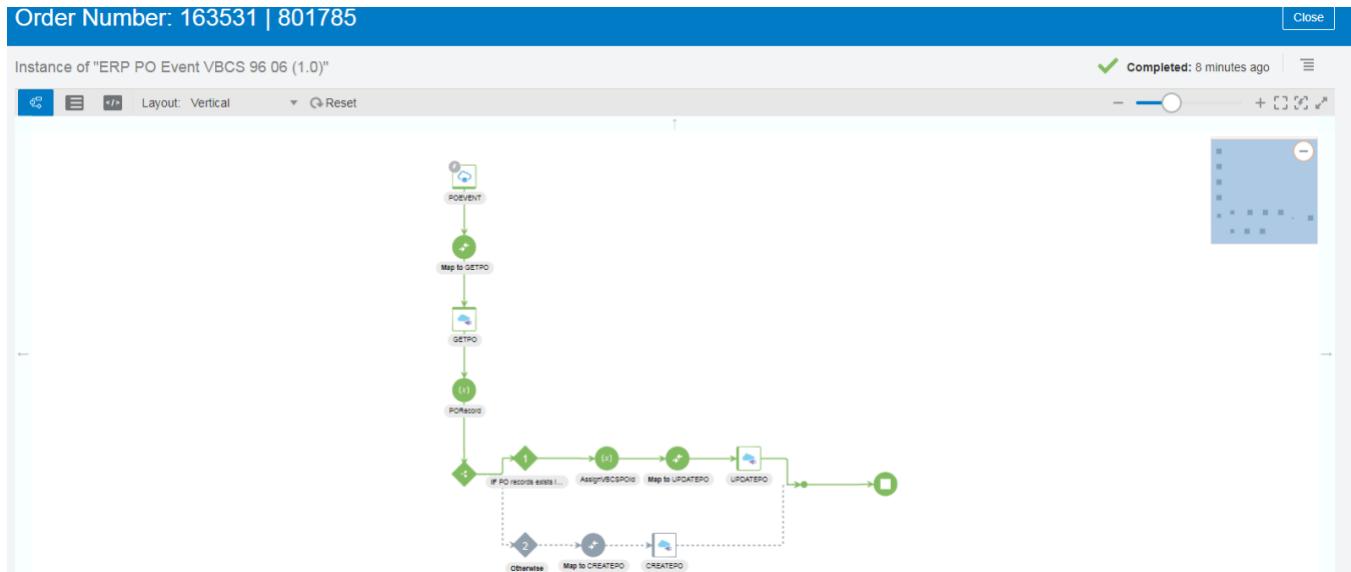
Cancel Change Order Back

11. In the Integrations -> Monitoring -> Tracking page, you can observe another instance got created. As, LOCID is got updated in ERP Cloud Flow 1 is triggered again.

Note: Purchase Order Event is raised both for Create and Update records

Instance	Order Number	Instance ID	Flow	Received	Completed	Duration
Flow1	163531	801785	Flow1	1 minute ago	1 minute ago	02 sec
Flow2	163531	801784	Flow2	4 minutes ago	3 minutes ago	33 sec

Open the Instance and we see that "If" path is evaluated to Update the PO into VBCS



12. Now go back to the PO List page. It should now show the "LOCId" updated.

POHeaderId	300000176127931
OrderNumber	163531
Supplier	Dell Inc.
LOCId	4

Congratulations, you have now completed Flow 2 and an end-to-end Flow to capture PO Events and update with cross reference of Letter of Credit Identifier from VBCS Web app.

